

FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY  
UNIVERZITY KOMENSKÉHO  
V BRATISLAVE

PÍ SOMNÁ PRÁCA  
K DIZERTAČNEJ SKÚŠKE

2005

Zuzana Holeščáková

FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY  
UNIVERZITY KOMENSKÉHO  
V BRATISLAVE

**Prediktor - korektor metódy vnútorného bodu  
riešenia úloh lineárneho programovania**

PÍ SOMNÁ PRÁCA K DIZERTAČNEJ SKÚŠKE

Zuzana Holeščáková

školiťka: Doc. RNDr. Margaréta Halická, CSc.

Bratislava 2005

# Obsah

<b>Značenie</b>	<b>2</b>
<b>1 Úvod</b>	<b>4</b>
<b>2 Základné pojmy</b>	<b>7</b>
2.1 Lineárne programovanie . . . . .	7
2.2 Primárno–duálne metódy vnútorného bodu . . . . .	8
2.3 Centrálna trajektória . . . . .	10
<b>3 Metódy sledovania centrálnej trajektórie</b>	<b>13</b>
3.1 Metódy s krátkym krokom (KK) . . . . .	15
3.1.1 Algoritmus KK . . . . .	15
3.2 Prediktor–korektor (PK) metódy . . . . .	24
3.2.1 Algoritmus PK . . . . .	25
3.3 Metódy s dlhým krokom (DK) . . . . .	30
3.3.1 Algoritmus DK . . . . .	31
<b>4 Mehrotrov algoritmus</b>	<b>36</b>
4.1 Nepripustné štartujúce body . . . . .	36
4.2 Charakteristika Mehrotrovho algoritmu . . . . .	37
4.3 Popis Mehrotrovho algoritmu . . . . .	39
4.3.1 Mehrotrov algoritmus . . . . .	41
<b>5 Projekt dizertačnej práce</b>	<b>42</b>
5.1 Súčasný stav problematiky . . . . .	42
5.2 Prehľad doktorandom dosiahnutých výsledkov . . . . .	45
5.3 Ciele dizertačnej práce . . . . .	46
<b>Literatúra</b>	<b>47</b>

## Značenie

$\mathbb{R}^n$	priestor reálnych $n$ -rozmerných vektorov
$\mathbb{R}^{(m \times n)}$	množina reálnych $(m \times n)$ rozmerných matíc
$A$	matica ( $\in \mathbb{R}^{m \times n}$ ) koeficientov lineárneho programovania
$x$	vektor ( $\in \mathbb{R}^n$ ) primárnych premenných
$y$	vektor ( $\in \mathbb{R}^m$ ) duálnych premenných
$s$	vektor ( $\in \mathbb{R}^n$ ) slackových premenných duálnej úlohy, $s = c - A^T y, s \geq 0$
$x^*$	optimálne riešenie primárnej úlohy (P)
$\mathcal{P}^*$	množina optimálnych riešení úlohy (P)
$(y^*, s^*)$	optimálne riešenie duálnej úlohy (D)
$\mathcal{D}^*$	množina optimálnych riešení úlohy (D)
$\mathcal{P}$	množina prípustných riešení primárnej úlohy (P)
$\mathcal{D}$	množina prípustných riešení duálnej úlohy (D)
$\mathcal{P}^o$	množina ostro prípustných riešení primárnej úlohy (P)
$\mathcal{D}^o$	množina ostro prípustných riešení duálnej úlohy (D)
$(x, y, s)$	premenné v primárno-duálnej úlohe
$(x^*, y^*, s^*)$	optimálne riešenie primárno-duálnej úlohy
$\mathcal{F}$	množina prípustných riešení primárno-duálnej úlohy
$\mathcal{F}^o$	množina ostro prípustných riešení primárno-duálnej úlohy
$\mathcal{F}^*$	množina optimálnych riešení primárno-duálnej úlohy
$X$	diagonálna matica pozostávajúca z prvkov vektora $x$ $X = \text{diag}(x_1, x_2, \dots, x_n)$
$S$	diagonálna matica pozostávajúca z prvkov vektora $s$ $S = \text{diag}(s_1, s_2, \dots, s_n)$
$e$	$n$ -rozmerný jednotkový stĺpcový vektor $e = (1, 1, \dots, 1)^T$
$F(x, y, s)$	funkcia $F(x, y, s) : \mathbb{R}^{2n+m} \rightarrow \mathbb{R}^{2n+m}$ pozostávajúca z KKT podmienok optimality
$J(x, y, s)$	Jakobiho matica funkcie $F(x, y, s)$
$(\Delta x, \Delta y, \Delta s)$	smerový vektor jednotlivých iterácií
$k$	index iterácie, $k = 0, 1, 2, \dots$
$(x^k, y^k, s^k)$	$k$ -ta iterácia primárno-duálnej metódy
$(\Delta x^k, \Delta y^k, \Delta s^k)$	smerový vektor $k$ -tej iterácie
$\alpha$	parameter dĺžky kroku, $\alpha \in (0, 1]$
$(x(\alpha), y(\alpha), s(\alpha))$	$(x, y, s) + \alpha(\Delta x, \Delta y, \Delta s)$

$\mu$	parameter centrálnnej trajektórie, $\mu > 0$
$\mathcal{C}$	centrálna trajektória $\mathcal{C} = \{(x(\mu), y(\mu), s(\mu)) \mid \mu > 0\}$
$\tau$	centrujúci (rozhodujúci) paramater, $\tau \in [0, 1]$
$\sigma$	mierka presnosti duality, $\sigma(x, s) = \frac{x^T s}{n}$
$\ \cdot\ _2$	Euklidovská norma, pre $u \in \mathbb{R}^n$ : $\ u\ _2 = (\sum_{i=1}^n u_i^2)^{1/2}$
$\ \cdot\ _1$	jednotková norma, pre $u \in \mathbb{R}^n$ : $\ u\ _1 = \sum_{i=1}^n  u_i $
$\mathcal{N}_2(\theta)$	okolie centrálnnej trajektórie, definované ako: $\left\{ (x, y, s) \in \mathcal{F}^o \mid \left\  \frac{XSe}{\sigma(x,s)} - e \right\ _2 \leq \theta \right\}$ , pre $\theta \in (0, 1)$ .
$\mathcal{N}_{-\infty}(\gamma)$	okolie centrálnnej trajektórie, definované ako: $\left\{ (x, y, s) \in \mathcal{F}^o \mid \frac{x_i s_i}{\sigma(x,s)} \geq \gamma, \quad \forall i = 1, 2, \dots, n \right\}$ , pre $\gamma \in (0, 1)$ .
$\epsilon$	parameter presnosti riešenia, $\epsilon > 0$
$r_b$	primárne reziduum, definované ako: $Ax - b$
$r_c$	duálne reziduum, definované ako: $A^T y + s - c$

# Kapitola 1

## Úvod

Lineárne programovanie, ako špecifický prípad konvexného matematického programovania, sa vyvíjalo samostatne a nezávisle od všetkých odvetví matematického programovania. Dantzigom (1947) navrhnutá simplexová metóda pre riešenie úloh lineárneho programovania bola totiž dlhú dobu považovaná za univerzálnu, spoľahlivú a výkonnú metódu, takže nebol žiaden praktický dôvod na hľadanie diametrálne iných postupov riešenia.

Lineárne programovanie našlo svoj rozmer predovšetkým v ekonomických aplikáciách. S prudkým rozvojom výpočtovej techniky došlo prirodzene aj k rozvoju lineárneho programovania a jeho aplikácií a otvárali sa tak nové možnosti riešiť aj úlohy veľkých rozmerov. Zároveň vznikol nový odbor, ktorý sa začal zaoberať výpočtovou zložitou, pričom za efektívne a rýchle programy sa považovali programy s polynomiálnou zložitou, t.j. že počet aritmetických operácií potrebných pre vyriešenie úlohy rozmeru  $m$  sa dá odhadnúť zhora polynómom v premennej  $m$ . Klee a Minty [12] (1972) ukázali, že simplexová metóda nie je polynomiálna a v najhorších prípadoch môže vyžadovať až exponenciálne veľa iterácií. Táto skutočnosť vhodne pripravila atmosféru pre hľadanie a zisťovanie alternatívnych postupov na riešenie úloh lineárneho programovania.

Nový priestor pre optimalizačné metódy lineárneho programovania, známe ako *metódy vnútorného bodu*, otvoril vo svojej práci Karmarkar [11] (1984), kde publikoval tzv. projektívny algoritmus pre lineárne programovanie, ktorý bol polynomiálny. Prepojenie Karmarkarovho algoritmu s metódami vnútorných bodov našiel Gill a kol. [4], keď ukázali úzky súvis algoritmu s logaritmicou bariérovou funkciou metód vnútorného bodu, ktorá bola svojho času neúspešne aplikovaná na úlohy nelineárneho programovania. Tieto skutočnosti upútali pozornosť matematickej obce a došlo tak za posledných dvadsať rokov k rozvoju ďalšej oblasti optimalizačných metód, novej éry metód vnútorného bodu, ktoré sú v súčasnosti jednou z najdiskutovanejších oblastí matematického programovania.

Karmarkarova práca bola značným prínosom a podnietila mnohých autorov k navrhovaniu početných algoritmov pre úlohy lineárneho programovania a možno povedať, že svojím spôsobom spôsobila revolúciu v optimalizácii. Ako sa ukázalo, najväčšiu popularitu získali primárno–duálne algoritmy sledovania centrálnej trajektórie, ktoré sa vyznačujú vysokým stupňom spoľahlivosti a jednoduchou štruktúrou.

Dôraz sa kládol na dôkazy polynomiálnej zložitosti, rýchlosti konvergenzie algoritmov a mnoho autorov sa pokúšalo taktiež o praktickú implementáciu jednotlivých algoritmov. Ako sa ukázalo, metódy vnútorného bodu sú veľmi efektívne najmä pre riešenie úloh veľkých rozmerov, a to vďaka svojej jednoduchosti a porovnateľne rýchlejšej konvergencii. Z praktického hľadiska bolo vyvinuté množstvo zdrojových kódov, z ktorých je dnes veľká časť voľne prístupná na rôznych webových stránkach venujúcich sa tejto problematike.

Neskôr sa tiež ukázalo, že navrhnuté algoritmy môžu byť zovšeobecnené aj pre úlohy kvadratického (napr. [22], [23]) a semidefinitného programovania (napr. [15]), bez toho, aby stratili zo svojej jednoduchosti a efektívnosti.

Hodnotý príspevok k problematike metód vnútorného bodu predložili vo svojej práci Lusitg, Marsten a Shanno [16], ktorý predostrel problém fungovania algoritmov aj pre neprípustné štartovacie body.

Veľmi prínosnou prácou bolo predstavenie Mehrotrovho prediktor–korektor algoritmu [19] (1992), ktorý sa stal základom pre väčšinu zdrojových kódov programov metód vnútorných bodov.

Mnoho tvrdení pre metódy vnútorného bodu sa už podarilo dokázať, niektoré oblasti však zostávajú stále nevyjasnené a nemajú teoretický matematický základ, akými sú napríklad globálna konvergencia alebo polynomiálna zložitosť pre Mehrotrov algoritmus. V súčasnosti je už vydaných aj zopár knižných publikácií, ktoré sa zaoberajú metódami vnútorných bodov. Medzi ne môžeme radiť napríklad aj monografie [1], [31], [33]. Prehľad o histórii a vývoji metód vnútorného bodu možno nájsť napríklad v [7], alebo [32].

Predkladaná práca je inšpirovaná najmä knihou Stephena J. Wrighta [33], ktorá je venovaná primárno–duálnym metódam vnútorného bodu. Pokúsime sa v nej sprostredkovať prehľad o metódach sledovania centrálnej trajektórie, ktoré tvoria jednu z dôležitých a v praxi najviac používaných metód vnútorného bodu. Budeme sa venovať hlavne Algoritmu KK (s krátkym krokom), Algoritmu PK (prediktor–korektor), Algoritmu DK (s dlhým krokom) a načrtneme problematiku vysoko populárneho Mehrotrovho algoritmu, kde zostáva ešte stále mnoho vecí neobjasnených. Oproti výkladu z [33], v tejto práci zovšeobecníme niektoré parametre používané v tzv. Algoritme PK (Kapitola 3.2) a celkovo podrobnejšie zdôvodníme a vyargumentujeme jednotlivé tvrdenia.

**Kapitola 2** prináša prehľad základných pojmov a definícií a taktiež predstavuje úvod do problematiky úloh lineárneho programovania, metód vnútorného bodu a definuje nám pojem centrálnej trajektórie. **Kapitola 3** pojednáva o metódach sledovania centrálnej trajektórie, prináša prehľad o jednotlivých metódach a ich algoritmoch. Pri uvádzaných algoritmoch uvažujeme prípustnosť štartujúceho bodu. Mehrotrovmu algoritmu je venovaná **Kapitola 4**, kde je vysvetlená koncepcia tohoto algoritmu. Metóda je prezentovaná vo všeobecnejšom tvare a teda berie do úvahy aj neprípustné štartovacie body. Aj napriek tomu, že je tento algoritmus prakticky najčastejšie používaný v zdrojových kódach programov pre metódy vútorného bodu a vyznačuje sa dobrými praktickými výsledkami, z teoretického hľadiska preň ostáva stále množstvo otvorených otázok. **Kapitola 5** dáva prehľad o súčasnom stave problematiky, doposiaľ dosiahnutých výsledkoch a otvára otázky a podnety pre ďalšie skúmanie a implementáciu algoritmov na triedy optimalizačných úloh z finančnej oblasti, ktorými sa chceme zaoberať neskôr v dizertačnej práci.



# Kapitola 2

## Základné pojmy

### 2.1 Lineárne programovanie

Uvažujme štandardnú formuláciu *primárnej úlohy* (P) a k nej prislúchajúcej *duálnej úlohy* (D) lineárneho programovania:

$$\begin{aligned} \text{(P)} \quad & \min\{c^T x \mid Ax = b, x \geq 0\} \\ \text{(D)} \quad & \max\{b^T y \mid A^T y + s = c, s \geq 0\}, \end{aligned}$$

pričom  $A \in \mathbb{R}^{m \times n}$  je matica typu  $(m \times n)$ , hodnosť matice  $A$  je  $m (\leq n)$ ,  $c, x, s \in \mathbb{R}^n$  sú  $n$ -rozmerné vektory,  $b, y \in \mathbb{R}^m$  sú  $m$ -rozmerné vektory.

Množinu optimálnych riešení  $x^*$  úlohy (P) označujeme  $\mathcal{P}^*$  a množinu optimálnych riešení  $(y^*, s^*)$  úlohy (D) označujeme  $\mathcal{D}^*$ .

Množinu  $\mathcal{P} = \{x \mid Ax = b, x \geq 0\}$  nazývame množinou prípustných riešení primárnej úlohy.

Množinu  $\mathcal{D} = \{(y, s) \mid A^T y + s = c, s \geq 0\}$  nazývame množinou prípustných riešení duálnej úlohy.

Množinu  $\mathcal{P}^o = \{x \mid Ax = b, x > 0\}$  nazývame množinou ostro prípustných riešení primárnej úlohy.

Množinu  $\mathcal{D}^o = \{(y, s) \mid A^T y + s = c, s > 0\}$  nazývame množinou ostro prípustných riešení duálnej úlohy.

Vzťah medzi (P) a (D) úlohami lineárneho programovania skúma a vysvetľuje *teória duality*. Pre stručné priblíženie uvedme (bez dôkazov) základné vety teórie duality:

**Veta 2.1 (Slabá veta o dualite)** *Ak  $x \in \mathcal{P}$  a  $(y, s) \in \mathcal{D}$ , tak potom  $b^T y \leq c^T x$ .*

Rozdiel  $c^T x - b^T y = x^T s$  nazývame duálnou medzerou v prípustnom bode  $(x, y, s)$ .

**Veta 2.2 (Silná veta o dualite)** *Optimálne riešenie  $x^*$  úlohy (P) existuje práve vtedy, keď existuje optimálne riešenie  $(y^*, s^*)$  úlohy (D), pričom  $b^T y^* = c^T x^*$ .*

Viac informácií možno nájsť napríklad v [31].

Nutné a postačujúce *podmienky optimality* pre každú z úloh (P) a (D) možno napísať v tvare spoločnom pre obe úlohy. Ide v podstate o známe *Karush–Kuhn–Tuckerove*<sup>1</sup> (KKT) podmienky z nelineárneho programovania [9], aplikované na úlohy (P) a (D):

$$Ax = b \quad (2.1)$$

$$A^T y + s = c \quad (2.2)$$

$$x_i s_i = 0, \quad i = 1, 2, \dots, n \quad (2.3)$$

$$(x, s) \geq 0. \quad (2.4)$$

Podmienky (2.1) a (2.4) určujú množinu prípustných riešení (P) a podmienky (2.2) a (2.4) určujú množinu prípustných riešení (D). Rovnicu (2.3) nazývame podmienkou komplementarity.

Vektor  $x^*$  je optimálne riešenie (P) a vektor  $(y^*, s^*)$  je optimálne riešenie (D) práve vtedy,  $(x, y, s) = (x^*, y^*, s^*)$  je riešením systému (2.1) – (2.4). Vektor  $(x^*, y^*, s^*)$  predstavuje *primárno–duálne optimálne riešenie*.

## 2.2 Primárno–duálne metódy vnútorného bodu

*Primárno–duálnymi metódami* súčasne hľadáme primárno–duálne riešenie úloh (P) a (D) v tvare  $(x^*, y^*, s^*)$ .

Množinu  $\mathcal{F} = \{(x, y, s) \mid Ax = b, A^T y + s = c, (x, s) \geq 0\}$  nazývame množinou prípustných riešení primárno–duálnej úlohy.

Množinu  $\mathcal{F}^o = \{(x, y, s) \mid Ax = b, A^T y + s = c, (x, s) > 0\}$  nazývame množinou ostro prípustných riešení primárno–duálnej úlohy.

<sup>1</sup>V literatúre sa zvyknú uvádzať aj pod názvom *Kuhn–Tuckerove* podmienky.

Množinu primárno–duálnych optimálnych riešení označíme  $\mathcal{F}^*$  môžeme ju zapísať v tvare:

$$\mathcal{F}^* = \mathcal{P}^* \times \mathcal{D}^* = \{(x^*, y^*, s^*) \mid \text{splňa podmienky optimality (2.1) – (2.4)}\}.$$

**Veta 2.3 (Veta o ostrej komplementarite)** *Ak  $\mathcal{P} \neq \emptyset$  a  $\mathcal{D} \neq \emptyset$ , potom existuje ostro komplementárne optimálne riešenie  $(x^*, y^*, s^*)$ , t.j. optimálne riešenie splňajúce  $x^* + s^* > 0$ .*

Veta o ostrej komplementarite zaručuje, že ak sú množiny prípustných riešení (P) a (D) neprázdne, tak existuje aspoň jedno ostro komplementárne riešenie  $(x^*, y^*, s^*)$ . Poznamenajme, že ak má úloha viac ako jedno optimálne riešenie, tak niektoré optimálne riešenia nie sú ostrokomplementárne.

Pomocou funkcie  $F(x, y, s) : \mathbb{R}^{2n+m} \rightarrow \mathbb{R}^{2n+m}$  a podmienky (2.4) prepíšeme podmienky optimality pre primárno–duálne metódy do tvaru:

$$F(x, y, s) = \begin{bmatrix} Ax - b \\ A^T y + s - c \\ XSe \end{bmatrix} = 0 \quad (2.5)$$

$$(x, s) \geq 0,$$

kde

$$\begin{aligned} X &= \text{diag}(x_1, x_2, \dots, x_n), \\ S &= \text{diag}(s_1, s_2, \dots, s_n), \\ e &= (1, 1, \dots, 1)^T. \end{aligned}$$

Ak by sme hypoteticky riešili tento systém pomocou Newtonovej metódy, tak by sme generovali postupnosť bodov  $(x^k, y^k, s^k)$  vo vnútri množiny prípustných riešení (t.j.  $x^k > 0$  a  $s^k > 0$ ), pomocou riešení  $(\Delta x, \Delta y, \Delta s)$  nasledovného systému:

$$J(x, y, s) \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = -F(x, y, s),$$

kde  $J(x, y, s)$  je Jakobiho matica funkcie  $F(x, y, s)$ .

To znamená, že by sme riešili systém:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -XSe \end{bmatrix} \quad (2.6)$$

v bode  $x = x^k$ ,  $s = s^k$ .

Nová iterácia v Newtonovom smere k optimálnemu primárno–duálnemu riešeniu by potom mala tvar:

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

pričom  $\alpha \in (0, 1]$  je parameter dĺžky kroku. Ak je  $\alpha = 1$ , hovoríme o plnom Newtonovom kroku. Tento krok by však bol zväčša neprípustný, lebo by ohrozil splnenie podmienky  $(x^{k+1}, s^{k+1}) > 0$ . Poznamenajme, že takéto postupy našli svoj obraz v tzv. afínno–škálovacích algoritmoch metód vnútorného bodu. Ich nevýhodou je, že jednotlivé iteračné body sa môžu už po pár iteráciách dostať pomerne blízko ku hranici kladného ortantu, čo spôsobuje, že musíme voliť  $\alpha \ll 1$ , aby sme zabezpečili kladnosť jednotlivých iterácií. Táto skutočnosť veľmi spomaľuje konvergenciu algoritmu. Preto sa v metódach vnútorného bodu smery určené systémom (2.6) – teda tzv. *afínno–škálovacie smery* (alebo inak povedané smery orientované k optimálnemu riešeniu) – kombinujú so smermi, ktoré nás orientujú viac do "centra" kladného ortantu. K definovaniu takýchto smerov slúži pojem centrálnej trajektórie.

## 2.3 Centrálna trajektória

Perturbujme systém (2.1)–(2.4) pre nejakú pevne zvolenú hodnotu  $\mu > 0$ :

$$Ax = b \tag{2.7}$$

$$A^T y + s = c \tag{2.8}$$

$$XSe = \mu e \tag{2.9}$$

$$(x, s) > 0. \tag{2.10}$$

Ak je množina  $\mathcal{F}^\circ$  neprázdna, platí, že pre každé  $\mu > 0$  existuje jednoznačné riešenie  $(x(\mu), y(\mu), s(\mu))$  perturbovaného systému (2.7)–(2.10).

*Centrálnu trajektóriu* môžeme potom definovať ako množinu bodov riešenia systému (2.7)–(2.10), závislých od hodnoty parametra  $\mu$ :

$$\mathcal{C} = \{(x(\mu), y(\mu), s(\mu)) \mid \mu > 0\}.$$

Pre zjednodušenie značenia budeme ďalej uvádzať namiesto  $(x(\mu), y(\mu), s(\mu))$  len značenie  $(x_\mu, y_\mu, s_\mu)$ .

V teórii metód vnútorného bodu sa dokázalo, že centrálna trajektória je dobre definovaná, t.j. že ak je množina ostro prípustných bodov  $\mathcal{F}^\circ$  neprázdna, tak má

systém (2.7)–(2.10) pre každé  $\mu > 0$  práve jedno riešenie  $(x_\mu, y_\mu, s_\mu)$ . Navyše centrálna trajektória pre  $\mu \rightarrow 0$  konverguje k ostro komplementárnemu optimálnemu riešeniu lineárneho programovania, t.j. platí:  $(x_\mu, y_\mu, s_\mu) \rightarrow (x^*, y^*, s^*)$ , pre  $\mu \rightarrow 0$ . Pre viac informácií pozri [33].

Pomocou funkcie  $F(x, y, s)$  z (2.5) možno centrujúci systém zapísať nasledovne:

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

Newtonov smer pre centrujúci systém (2.7)–(2.9) tak dostaneme riešením systému:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -XSe + \mu e \end{bmatrix}. \quad (2.11)$$

Tento smer je orientovaný k centrálnej trajektórii.

Ak by sme systém (2.11) postupne riešili pre zmenšujúce sa  $\mu$ , takýmto riešením by sme dospeli k optimálnemu riešeniu pôvodnej neperturbovanej úlohy a to bez ohľadu na skutočnosť, v akom bode sa nachádzame.

Väčšina primárno–duálnych algoritmov používa najčastejšie kombináciu Newtonovho kroku orientovaného smerom k optimálnemu primárno–duálnemu riešeniu z (2.6) spolu s Newtonovým krokom orientovaným smerom k centrálnej trajektórii z (2.11). V algoritmoch, ktoré popíšeme nižšie, budeme hodnotu paramtera  $\mu$  voliť adaptívne, v závislosti od toho, v akom bode sa práve nachádzame. Pri tomto budeme používať dva ďalšie parametre  $\tau$  a  $\sigma(x, s)$ , kde  $\mu = \tau \sigma(x, s)$ , pričom  $\tau \in [0, 1]$  bude *centrujúci (rozhodujúci) parameter* a  $\sigma(x, s)$  bude *parameter duálnej medzery*, ktorý je daný predpisom:

$$\sigma(x, s) = \frac{1}{n} \sum_{i=1}^n x_i s_i = \frac{x^T s}{n}. \quad (2.12)$$

V ďalšom texte budeme kvôli zjednodušeniu  $\sigma(x, s)$  zväčša označovať iba symbolom  $\sigma$ , pričom pod týmto označením budeme stále chápať funkciu závislú od premenných  $x$  a  $s$ .

Riešením systému:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -XSe + \tau \sigma e \end{bmatrix} \quad (2.13)$$

dostávame Newtonov smer  $(\Delta x, \Delta y, \Delta s)$  orientovaný k bodu  $(x_{\tau\sigma}, y_{\tau\sigma}, s_{\tau\sigma}) \in \mathcal{C}$ .

Pomocou voľby centrujúceho parametra  $\tau$  rozhodujeme o smere Newtonovho kroku. Pre hodnotu parametra  $\tau = 0$  dostávame *afínno-škálovací smer*, orientovaný k optimálnemu primárno-duálnemu riešeniu. Ak položíme  $\tau = 1$ , dostávame *centrujúci smer*, pri ktorom Newtonov krok smeruje k bodu  $(x_\sigma, y_\sigma, s_\sigma) \in \mathcal{C}$ . Väčšina algoritmov však používa voľbu centrujúceho parametra  $\tau \in (0, 1)$ .

Na základe týchto poznatkov môžeme sformulovať základnú štruktúru primárno-duálneho algoritmu, ktorý bude východiskom pre ďalšie modifikácie v metódach sledovania centrálnej trajektórie.

### Skelet algoritmu

**Vstup:**  $(x^o, y^o, s^o) \in \mathcal{F}^o$

pre  $k = 0, 1, 2, \dots$

#### 1. rieš:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -X^k S^k e + \tau_k \sigma_k e \end{bmatrix}, \quad (2.14)$$

kde  $\tau_k \in [0, 1]$  a  $\sigma_k = \frac{(x^k)^T s^k}{n}$ ;

#### 2. prirad:

$$(x^{k+1}, y^{k+1}, s^{k+1}) \leftarrow (x^k, y^k, s^k) + \alpha_k (\Delta x^k, \Delta y^k, \Delta s^k), \quad (2.15)$$

kde  $\alpha_k$  volíme tak, aby  $(x^{k+1}, s^{k+1}) > 0$ .

**skonči.**

# Kapitola 3

## Metódy sledovania centrálnej trajektórie

Metódy sledovania centrálnej trajektórie zabezpečujú, aby sa jednotlivé iterácie nachádzali v okolí centrálnej trajektórie a aby sa pozdĺž nej, so zmenšujúcim sa  $\mu$  hľadalo optimálne riešenie úlohy lineárneho programovania. Pridržiavať sa okolia centrálnej trajektórie má tú výhodu, že toto okolie je vzdialené od bodov, ktoré sa nachádzajú na hranici nezáporného ortantu. Nie je nutné, aby sme sa nachádzali priamo na  $\mathcal{C}$ , alebo v jej tesnej blízkosti. To znamená, že podmienka (2.9) z KKT podmienok nemusí byť presne splnená. Postačí, ak sa budeme nachádzať v korektno definovanom okolí centrálnej trajektórie, pričom sa budeme snažiť znižovať parameter presnosti mierky duality  $\sigma$  k nule.

Definujme okolia centrálnej trajektórie  $\mathcal{C}$ , ako:

$$\mathcal{N}_2(\theta) = \left\{ (x, y, s) \in \mathcal{F}^o \mid \left\| \frac{XSe}{\sigma(x, s)} - e \right\|_2 \leq \theta \right\} \quad (3.1)$$

pre parameter  $\theta \in (0, 1)$  a

$$\mathcal{N}_{-\infty}(\gamma) = \left\{ (x, y, s) \in \mathcal{F}^o \mid \frac{x_i s_i}{\sigma(x, s)} \geq \gamma, \quad \forall i = 1, 2, \dots, n \right\} \quad (3.2)$$

pre parameter  $\gamma \in (0, 1)$ .

Označenie  $\|\cdot\|_2$  predstavuje *Euklidovskú normu*, ktorá je vo všeobecnosti pre vektor  $u \in \mathbb{R}^n$  definovaná predpisom:

$$\|u\|_2 = \left( \sum_{i=1}^n u_i^2 \right)^{1/2}.$$

Metódy sledovania centrálnej trajektórie možno z časti prirovnať k homotopickým metódam, ktoré taktiež hľadajú riešenie pozdĺž trajektórie závislej od ne-

jakého parametra, ktorý môže zdanlivo neriešiteľný problém zmeniť na triviálny.

V metódach sledovania centrálnej trajektórie volíme okolie  $\mathcal{C}$ ,<sup>1</sup> ďalej centrujúci parameter  $\tau$  a dĺžku kroku  $\alpha$ , a to všetko tak, aby sme zabezpečili, že všetky iterácie  $(x^k, y^k, s^k)$  sa budú nachádzať v zvolenom okolí.

Pri odhade zložitosti algoritmov sa často hovorí o takzvanom  $\epsilon$ -presnom riešení.  $\epsilon$ -presné riešenie je také  $x \in \mathcal{P}$  a  $s \in \mathcal{D}$ , ktoré spĺňa podmienku  $x^T s < \epsilon$ . V algoritmoch opakujeme cyklus iterácií až pokiaľ nie je splnená požadovaná podmienka presnosti.

Metódy, v ktorých sa používa voľba okolia  $\mathcal{N}_2(\theta)$  majú zložitosť  $O(\sqrt{n} \log \frac{1}{\epsilon})$ . Medzi takéto patria metódy s krátkym krokom a prediktor–korektor metódy.

*Metódy s krátkym krokom* patria k najjednoduchším metódam vnútorného bodu. Pri voľbe centrujúceho parametra  $\tau$  sa správajú dosť konzervatívne. Pre všetky iterácie volia za parameter  $\tau_k \equiv \tau$  konštantnú hodnotu, zväčša blízku jednotke a za parameter dĺžky kroku  $\alpha_k \equiv 1$ .

*Prediktor–korektor metódy*<sup>2</sup> sú ďalšími z kategórie metód sledovania centrálnej trajektórie so zložitou  $O(\sqrt{n} \log \frac{1}{\epsilon})$ . V porovnaní s ostatnými metódami dosahujú lepšie výsledky, napríklad čo sa týka rýchlosti konvergencie. V jednotlivých iteráciách sa striedajú dva druhy krokov a to *prediktor krok* ( $\tau = 0$ ), ktorý je orientovaný smerom k optimálnemu primárno–duálnemu riešeniu,<sup>3</sup> vylepšuje hodnotu parametra mierky presnosti duality  $\sigma$  a *korektor krok* ( $\tau = 1$  a  $\alpha = 1$ ), ktorý zabezpečí nasmerovanie späť do vnútra množiny, smerom k centrálnej trajektórii, bez ďalšieho efektu na hodnotu parametra  $\sigma$ .

*Metódy s dlhým krokom* používajú vo svojich algoritmoch okolie  $\mathcal{N}_{-\infty}(\gamma)$ . Ich zložitosť je  $O(n \log \frac{1}{\epsilon})$ , čo je v porovnaní so zložitou algoritmov predchádzajúcich dvoch metód,  $O(\sqrt{n} \log \frac{1}{\epsilon})$ , slabší výsledok. Vyznačujú sa odvážnejšou voľbou centrujúceho parametra  $\tau$  (napríklad v porovnaní s metódami s krátkym krokom). Voľbu parametra dĺžky kroku  $\alpha_k$  prispôsobuje vždy najväčšej možnej hodnote s ohľadom na to, aby sme sa stále nachádzali v zvolenom okolí.

Ďalej sa budeme bližšie zaoberať jednotlivými algoritmi. Algoritmy sa v globále vyznačujú jednoduchou štruktúrou, aj napriek ich zložitejšiemu teoretickému pozadiu. Spomínané algoritmy sú základom pre zložitejšie a komplikovanejšie algoritmy, akými sú napríklad algoritmy s neprípustným štartujúcim bodom.

<sup>1</sup>Druh okolia závisí od typu použitej normy. Nemusi teda ísť v každom prípade iba o uvedené dva druhy okolia  $\mathcal{N}_2(\theta)$  a  $\mathcal{N}_{-\infty}(\gamma)$ .

<sup>2</sup>*Prediktor–korektor algoritmus* sa niekedy označuje názvom ”Mizuno–Todd–Ye prediktor korektor algoritmus”, s cieľom odlíšiť ho od značne rozdielného ”Mehrotra prediktor–korektor algoritmu”.

<sup>3</sup>Orientáciu smerom k optimálnemu primárno–duálnemu riešeniu nazývame afinno–škálovacím smerom.



My však budeme v tejto kapitole predpokladať prípustnosť štartujúceho bodu. Poznamenajme, že nižšie uvedené metódy môžu byť modifikované aj pre prípad neprípustného štartujúceho bodu.

### 3.1 Metódy s krátkym krokom (KK)

V metódach s krátkym krokom (KK) štartujeme v bode  $(x^o, y^o, s^o) \in \mathcal{N}_2(\theta)$  a pre celý algoritmus používame konštanté hodnoty parametrov  $\alpha_k = 1$  a  $\tau_k = \tau$ , pričom volené parametre  $\theta$  a  $\tau$  musia spĺňať kritéria, ktorých analýze sa budeme ďalej podrobnejšie venovať. Na jednotlivé iterácie  $(x^k, y^k, s^k)$ , generované touto metódou, kladieme požiadavku, aby sa všetky nachádzali vo vnútri zvoleného okolia  $\mathcal{N}_2(\theta)$ . Ako tiež neskôr uvidíme, mierka presnosti duality  $\sigma_k = \sigma(x^k, s^k)$  konverguje k nule s konštantnou rýchlosťou  $\tau$ .

Táto metóda je definovaná na základe Skeletu algoritmu z Kapitoly 2.3 zo strany 12.

#### 3.1.1 Algoritmus KK

**Vstup:**  $\theta = 0,4$   
 $\tau = 1 - \frac{0,4}{\sqrt{n}}$   
 $(x^o, y^o, s^o) \in \mathcal{N}_2(\theta)$   
**pre**  $k = 0, 1, 2, \dots$

1. **prirad:**  $\tau_k = \tau$ ,

2. **rieš:**

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -X^k S^k e + \tau_k \sigma_k e \end{bmatrix}, \quad (3.3)$$

3. **prirad:**

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

**skonči.**

Keďže sme položili  $\alpha_k = 1$  pre všetky iterácie, neovplyvňovali sme tak dĺžku jednotlivých krokov v iteráciách.

Pri nasledujúcich analýzach metódy sledovania centrálnej trajektórie s krátkym krokom sa budeme snažiť ukázať, že všetky generované iterácie sa stále nachádzajú vo vnútri okolia  $\mathcal{N}_2(\theta)$ . Ešte predtým však dokážme globálnu konvergenciu a polynomiálnu zložitosť algoritmu. Vychádzajme zatiaľ z faktu, že platí tvrdenie, že sa všetky iterácie nachádzajú vo vnútri zvoleného okolia.

Z nasledujúcej Lemy 3.1 vyplýva lineárna konvergencia. Uvádzame ju vo všeobecnom tvare, pretože sa dá tak aplikovať na všetky druhy algoritmov, ktoré používajú na výpočet kroku  $(\Delta x^k, \Delta y^k, \Delta s^k)$  systém (3.3).

Budeme používať nasledovné značenie:

$$\begin{aligned} (x(\alpha), y(\alpha), s(\alpha)) &= (x, y, s) + \alpha(\Delta x, \Delta y, \Delta s), \\ \sigma(\alpha) &= \frac{x(\alpha)^T s(\alpha)}{n}. \end{aligned} \quad (3.4)$$

**Lema 3.1** *Nech je krok  $(\Delta x, \Delta y, \Delta s)$  definovaný systémom (3.3).*

*Potom*

$$\Delta x^T \Delta s = 0 \quad (3.5)$$

*a*

$$\sigma(\alpha) = (1 - \alpha(1 - \tau))\sigma. \quad (3.6)$$

*Dôkaz:* Najskôr dokážeme prvú časť vety, t.j. že platí rovnosť  $\Delta x^T \Delta s = 0$ . Vychádzajúc z prvého a druhého riadku systému (3.3) máme:

$$\begin{aligned} A\Delta x &= 0, \\ A^T \Delta y + \Delta s &= 0 \quad \Rightarrow \Delta s = -A^T \Delta y, \end{aligned}$$

z čoho je zrejماً platnosť:

$$\Delta x^T \Delta s = -\Delta x^T A^T \Delta y = -(A\Delta x)^T \Delta y = 0.$$

Pre druhú časť dôkazu budeme vychádzať z tretieho riadku systému (3.3):

$$S\Delta x + X\Delta s = -XSe + \tau\sigma e. \quad (3.7)$$

Sčítaním všetkých  $n$ -zložiek tejto rovnice dostávame:

$$\begin{aligned} s^T \Delta x + x^T \Delta s &= -x^T s + n\tau\sigma \\ &= -x^T s + n\tau \frac{x^T s}{n} \\ &= -(1 - \tau)x^T s \end{aligned}$$

Z tohoto a z rovnosti (3.5) ďalej dostávame:

$$\begin{aligned} x(\alpha)^T s(\alpha) &= x^T s + \alpha(s^T \Delta x + x^T \Delta s) + \alpha^2 \Delta x^T \Delta s \\ &= x^T s - \alpha(1 - \tau)x^T s \\ &= (1 - \alpha(1 - \tau))x^T s, \end{aligned}$$

z čoho jasne vyplýva druhá rovnosť (3.6).  $\square$

Týmto sa nám podarilo zároveň dokázať *globálnu lineárnu konvergenciu* Algoritmu KK, keďže pri špecifickej voľbe parametrov v algoritme,  $\tau_k = \tau = \left(1 - \frac{0,4}{\sqrt{n}}\right)$  a  $\alpha_k = 1$ , dostávame:

$$\sigma_{k+1} = \tau \sigma_k = \left(1 - \frac{0,4}{\sqrt{n}}\right) \sigma_k \quad k = 0, 1, \dots \quad (3.8)$$

Nasledujúcu vetu uvedme pre potreby dôkazu polynomiálnej zložitosti Algoritmu KK.

**Veta 3.1** *Nech  $\epsilon \in (0, 1)$ . Predpokladajme, že algoritmus pre riešenie systému (2.1)–(2.4) generuje postupnosť iterácií, pre ktorú platí*

$$\sigma_{k+1} \leq \left(1 - \frac{\delta}{n^\omega}\right) \sigma_k \quad k = 0, 1, \dots \quad (3.9)$$

pre nejaké kladné konštanty  $\delta$  a  $\omega$ .

Predpokladajme taktiež, že štartujúci bod  $(x^o, y^o, s^o)$  vyhovuje podmienke:

$$\sigma_o \leq \frac{1}{\epsilon^\kappa} \quad (3.10)$$

pre nejakú kladnú konštantu  $\kappa$ .

Potom existuje index  $K$ :

$$K = O\left(n^\omega \log \frac{1}{\epsilon}\right),$$

taký že

$$\sigma_k \leq \epsilon \quad \forall k \geq K.$$

*Dôkaz:* Je zřejmé, že opakovaným použitím (3.9) dostaneme:

$$\sigma_k \leq \left(1 - \frac{\delta}{n^\omega}\right)^k \sigma_o \quad k = 1, 2, \dots$$

Zlogaritmovaním tejto formuly a potom aplikovaním vaňahu (3.10) dostávame:

$$\begin{aligned} \log \sigma_k &\leq k \log \left(1 - \frac{\delta}{n^\omega}\right) + \log \sigma_o \\ &\leq k \log \left(1 - \frac{\delta}{n^\omega}\right) + \kappa \log \frac{1}{\epsilon}. \end{aligned}$$

Všeobecne, pre logaritmickú funkciu platí odhad:

$$\log(1 + \beta) \leq \beta, \quad \forall \beta > -1.$$

Rovnosť nastáva iba v prípade, ak je  $\beta = 0$ .

Na základe tohoto odhadu vyplýva, že:

$$\log \sigma_k \leq k \left( -\frac{\delta}{n^\omega} \right) + \kappa \log \frac{1}{\epsilon}.$$

Konvergenčné kritérium,  $\sigma_k \leq \epsilon$ , je preto splnené, ak máme:

$$k \left( -\frac{\delta}{n^\omega} \right) + \kappa \log \frac{1}{\epsilon} \leq \log \epsilon.$$

Úpravou výrazu dostávame

$$\begin{aligned} k \left( -\frac{\delta}{n^\omega} \right) &\leq -\log \frac{1}{\epsilon} - \kappa \log \frac{1}{\epsilon} \\ &= -(1 + \kappa) \log \frac{1}{\epsilon}, \end{aligned}$$

z čoho je zrejmé, že táto nerovnosť platí  $\forall k$ , ktoré spĺňajú podmienku:

$$k \geq K = (1 + \kappa) \frac{n^\omega}{\delta} \log \frac{1}{\epsilon}.$$

Týmto je veta dokázaná.  $\square$

Z nasledujúcej vety priamo vyplýva polynomiálna zložitosť Algoritmu KK.

**Veta 3.2** *Majme  $\epsilon > 0$ . Predpokladajme, že pre štartujúci bod  $(x^o, y^o, s^o) \in \mathcal{N}_2(0, 4)$  v Algoritme KK platí:*

$$\sigma_o \leq \frac{1}{\epsilon^\kappa}$$

pre nejaké  $\kappa > 0$ .

Potom existuje index  $K$ , kde  $K = O(\sqrt{n} \log \frac{1}{\epsilon})$ , taký že:

$$\sigma_k \leq \epsilon \quad \forall k \geq K.$$

*Dôkaz:* Je bezprostredným dôsledkom výrazu (3.8) a vety 3.1 pre voľbu parametrov  $\delta = 0,4$  a  $\omega = 0,5$ .  $\square$

Predchádzajúcimi tvrdeniami sme dokázali lineárnu konvergenciu a polynomiálnu zložitosť Algoritmu KK. Ďalej sa budeme venovať cieľu, ukázať že jednotlivé iterácie  $(x^k, y^k, s^k)$  všetky zostávajú vo vnútri okolia  $\mathcal{N}_2(\theta)$ .

Ako technický aparát pre ďalšie dôkazy uveďme nasledujúcu lemu.

**Lema 3.2** *Nech  $u$  a  $v$  sú ľubovoľné vektory z  $\mathbb{R}^n$ , pričom  $u^T v \geq 0$ . Potom platí:*

$$\|U V e\| \leq 2^{-3/2} \|u + v\|^2,$$

kde

$$U = \text{diag}(u_1, u_2, \dots, u_n), \quad V = \text{diag}(v_1, v_2, \dots, v_n).$$

*Dôkaz:* Pre akékoľvek dva skaláry  $\alpha$  a  $\beta$ , také že  $\alpha\beta \geq 0$ , platí:

$$\sqrt{\alpha\beta} \leq \frac{1}{2} |\alpha + \beta|, \quad (3.11)$$

čo je zrejmé z nasledovných úprav:

$$\begin{aligned} 2\sqrt{\alpha\beta} &\leq |\alpha + \beta| \\ 4\alpha\beta &\leq \alpha^2 + 2\alpha\beta + \beta^2 \\ \alpha^2 - 2\alpha\beta + \beta^2 &\geq 0 \\ (\alpha - \beta)^2 &\geq 0. \end{aligned}$$

Kedže  $u^T v \geq 0$ , dostávame:

$$0 \leq u^T v = \sum_{u_i v_i \geq 0} u_i v_i + \sum_{u_i v_i < 0} u_i v_i = \sum_{i \in \mathcal{K}} |u_i v_i| - \sum_{i \in \mathcal{L}} |u_i v_i|,$$

kde sme rozčlenili množinu indexov  $1, 2, \dots, n$  na

$$\mathcal{K} = \{i \mid u_i v_i \geq 0\} \quad \text{a} \quad \mathcal{L} = \{i \mid u_i v_i < 0\}.$$

Z toho dostávame, že:

$$\sum_{i \in \mathcal{L}} |u_i v_i| \leq \sum_{i \in \mathcal{K}} |u_i v_i|. \quad (3.12)$$

Všeobecne, definujme *jednotkovú normu* pre ľubovoľný vektor  $u \in \mathbb{R}^n$  predpisom:

$$\|u\|_1 = \sum_{i=1}^n |u_i|.$$

Ak vychádzame z toho, že pre normy platí  $\|\cdot\|_2 \leq \|\cdot\|_1$ , tak dostávame:

$$\begin{aligned} \|U V e\| &= \left( \|[u_i v_i]_{i \in \mathcal{K}}\|_1^2 + \|[u_i v_i]_{i \in \mathcal{L}}\|_1^2 \right)^{1/2} \\ &\leq \left( \|[u_i v_i]_{i \in \mathcal{K}}\|_1^2 + \|[u_i v_i]_{i \in \mathcal{L}}\|_1^2 \right)^{1/2} \quad \|\cdot\|_2 \leq \|\cdot\|_1 \\ &\leq \left( 2 \|[u_i v_i]_{i \in \mathcal{K}}\|_1^2 \right)^{1/2} \quad \text{podľa (3.12)} \end{aligned}$$

$$\begin{aligned}
 &= \sqrt{2} \left\| [u_i v_i]_{i \in \mathcal{K}} \right\|_1 \\
 &\leq \sqrt{2} \left\| \left[ \frac{1}{4} (u_i + v_i)^2 \right]_{i \in \mathcal{K}} \right\|_1 \quad \text{podľa (3.11), } i \in \mathcal{K} \\
 &= 2^{-3/2} \sum_{i \in \mathcal{K}} (u_i + v_i)^2 \\
 &\leq 2^{-3/2} \sum_{i=1}^n (u_i + v_i)^2 \\
 &= 2^{-3/2} \|u + v\|^2,
 \end{aligned}$$

čím sme dokázali platnosť lemy.  $\square$

V leme 3.1 sme ukázali, že vektorový súčin  $\Delta x^T \Delta s = \sum \Delta x_i \Delta s_i$  je rovný nule. Nie je však žiaden špeciálny dôvod na to, aby boli nulové jednotlivé súčiny  $\Delta x_i \Delta s_i$ .

V nasledujúcej leme budeme hľadať hranicu na vektor týchto dvojíc súčinov.

**Lema 3.3** *Ak  $(x, y, s) \in \mathcal{N}_2(\theta)$ , potom*

$$\|\Delta X \Delta S e\| \leq \frac{\theta^2 + n(1 - \tau)^2}{2^{3/2}(1 - \theta)} \sigma.$$

*Dôkaz:* Definujme diagonálnu maticu:  $D = X^{1/2} S^{1/2}$ .

Po vynásobení vztáhu (3.7) výrazom  $(XS)^{-1/2}$  dostávame:

$$D^{-1} \Delta x + D \Delta s = (XS)^{-1/2} (-XSe + \tau \sigma e). \quad (3.13)$$

Aplikovaním predchádzajúcej Lemy 3.2, kde v našom prípade  $u = D^{-1} \Delta x$  a  $v = D \Delta s$ , dostávame:<sup>4</sup>

$$\begin{aligned}
 \|\Delta X \Delta S e\| &= \|(D^{-1} \Delta X)(D \Delta S) e\| \\
 &\leq 2^{-3/2} \|D^{-1} \Delta X + D \Delta S\|^2 \quad \text{z Lemy 3.2} \\
 &= 2^{-3/2} \|(XS)^{-1/2} (-XSe + \tau \sigma e)\|^2 \quad \text{z (3.13)} \\
 &= 2^{-3/2} \sum_{i=1}^n \frac{(-x_i s_i + \tau \sigma)^2}{x_i s_i} \\
 &\leq 2^{-3/2} \frac{\|XSe - \tau \sigma e\|^2}{\min_i x_i s_i}. \quad (3.14)
 \end{aligned}$$

Keďže  $(x, y, s) \in \mathcal{N}_2(\theta)$ , dostávame že

$$\min_i x_i s_i \geq (1 - \theta) \sigma. \quad (3.15)$$

<sup>4</sup>Pre našu voľbu vektorov  $u = D^{-1} \Delta x$  a  $v = D \Delta s$  je možné lemu aplikovať, keďže platí, že  $(D^{-1} \Delta x)^T (D \Delta s) = 0$ .

Pre ohraničenie výrazu  $\|XSe - \tau\sigma e\|$  poznamenajme, že:

$$e^T(XSe - \sigma e) = x^T s - \sigma e^T e = 0,$$

a preto:

$$\begin{aligned} \|XSe - \tau\sigma e\|^2 &= \|(XSe - \sigma e) + (1 - \tau)\sigma e\|^2 \quad / \pm \sigma e \\ &= \|XSe - \sigma e\|^2 + 2(1 - \tau)\sigma e^T(XSe - \sigma e) + \\ &\quad + (1 - \tau)^2\sigma^2 e^T e \\ &\leq \theta^2\sigma^2 + n(1 - \tau)^2\sigma^2. \end{aligned} \quad (3.16)$$

Dosadením (3.15) a predchádzajúceho vzťahu (3.16) do výrazu (3.14) dostávame:

$$\begin{aligned} \|\Delta X \Delta S e\| &\leq 2^{-3/2} \frac{\|XSe - \tau\sigma e\|}{\min_i x_i s_i} \\ &\leq 2^{-3/2} \frac{\theta^2\sigma^2 + n(1 - \tau)^2\sigma^2}{(1 - \theta)\sigma} \\ &= \frac{\theta^2 + n(1 - \tau)^2}{2^{3/2}(1 - \theta)}\sigma. \end{aligned}$$

Tým sme dokázali platnosť lemy.  $\square$

Na základe lemy 3.1 teraz už vieme, že parameter mierky duality  $\sigma$  sa znižuje lineárne, ako sa posúvame v smere  $(\Delta x, \Delta y, \Delta s)$ . Ako ďaleko sa sa nachádza bod  $(x(\alpha), y(\alpha), s(\alpha))$  od centrálnej trajektórie pri použití okolia  $\mathcal{N}_2(\theta)$  a normy  $\|\cdot\|_2$  objasňuje nasledujúca lema, ktorá je jednoduchým dôsledkom predchádzajúcej Lemy 3.3.

**Lema 3.4** *Ak  $(x, y, s) \in \mathcal{N}_2(\theta)$ , potom platí:*

$$\begin{aligned} \|X(\alpha)S(\alpha)e - \sigma(\alpha)e\| &\leq |1 - \alpha|\|XSe - \sigma e\| + \alpha^2\|\Delta X \Delta S e\| \\ &\leq |1 - \alpha|\theta\sigma + \alpha^2 \frac{\theta^2 + n(1 - \tau)^2}{2^{3/2}(1 - \theta)}\sigma. \end{aligned}$$

*Dôkaz:* Vychádzajúc z tretieho riadku systému (3.7) objasníme jednotlivé zložky vzťahu  $X(\alpha)S(\alpha)e - \sigma(\alpha)e$ . Na základe tohoto a z Lemy 3.1 potom dostávame:

$$\begin{aligned} x_i(\alpha)s_i(\alpha) - \sigma(\alpha) &= x_i s_i + \alpha(s_i \Delta x_i + x_i \Delta s_i) + \alpha^2 \Delta x_i \Delta s_i - \\ &\quad - (1 - \alpha(1 - \tau))\sigma \\ &= x_i s_i + \alpha(-x_i s_i + \tau\sigma) + \alpha^2 \Delta x_i \Delta s_i - \\ &\quad - (1 - \alpha(1 - \tau))\sigma \\ &= x_i s_i(1 - \alpha) + \alpha\tau\sigma + \alpha^2 \Delta x_i \Delta s_i - \\ &\quad - (1 - \alpha + \alpha\tau)\sigma \\ &= x_i s_i(1 - \alpha) + \alpha^2 \Delta x_i \Delta s_i - (1 - \alpha)\sigma. \end{aligned}$$

Prepísaním týchto zložiek do vektorovej formy dostávame:

$$\begin{aligned}
 \|X(\alpha)S(\alpha)e - \sigma(\alpha)e\| &= \|[x_i s_i(1 - \alpha) - (1 - \alpha)\sigma + \alpha^2 \Delta x_i \Delta s_i]_{i=1}^n\| \\
 &\leq |1 - \alpha| \|XSe - \sigma e\| + \alpha^2 \|\Delta X \Delta Se\| \\
 &\leq |1 - \alpha| \theta \sigma + \alpha^2 \frac{\theta^2 + n(1 - \tau)^2}{2^{3/2}(1 - \theta)} \sigma,
 \end{aligned}$$

čo bolo treba dokázať.  $\square$

Ďalšia veta, ktorú uvádzame, definuje vzťah medzi parametrami  $\theta$  a  $\tau$  a ukazuje, že aj pri voľbe kroku dĺžky  $\alpha = 1$  sa nedostaneme s novou iteráciou mimo okolia  $\mathcal{N}_2(\theta)$ .

**Veta 3.3** *Nech parametre  $\tau \in (0, 1)$  a  $\theta \in (0, 1)$  vyhovujú podmienke:*

$$\frac{\theta^2 + n(1 - \tau)^2}{2^{3/2}(1 - \theta)} \leq \tau \theta. \quad (3.17)$$

Potom, ak  $(x, y, s) \in \mathcal{N}_2(\theta)$ , tak

$$(x(\alpha), y(\alpha), s(\alpha)) \in \mathcal{N}_2(\theta) \quad \forall \alpha \in [0, 1].$$

*Dôkaz:* Ak dosadíme podmienku (3.17) do vzťahu z Lemy 3.4, pre  $\alpha \in [0, 1]$  máme:

$$\begin{aligned}
 \|X(\alpha)S(\alpha)e - \sigma(\alpha)e\| &\leq (1 - \alpha)\theta\sigma + \alpha^2\tau\theta\sigma \\
 &\leq (1 - \alpha + \alpha\tau)\theta\sigma \\
 &= (1 - \alpha(1 - \tau))\theta\sigma \\
 &= \theta\sigma(\alpha),
 \end{aligned} \quad (3.18)$$

kde poslednú rovnosť sme dostali zo vzťahu (3.6).

Z tohoto dôvodu spĺňa bod  $(x(\alpha), y(\alpha), s(\alpha))$  podmienku, že patrí okoliu  $\mathcal{N}_2(\theta)$ . Ešte nám zostáva overiť, že  $(x(\alpha), y(\alpha), s(\alpha)) \in \mathcal{F}^o$ .

Overme teda všetky podmienky, ktoré musí spĺňať bod patriaci do  $\mathcal{F}^o$ .

Je ľahké overiť splnenie podmienok:

$$\begin{aligned}
 Ax(\alpha) &= b \\
 A^T y(\alpha) + s(\alpha) &= c
 \end{aligned}$$

Na základe definície  $(x(\alpha), y(\alpha), s(\alpha))$  zo vzťahu (3.4) a prvého riadku systému (3.3) dostávame:

$$\begin{aligned}
 Ax(\alpha) &= A(x + \alpha\Delta x) \\
 &= Ax + \alpha A\Delta x \\
 &= b + \alpha 0 \\
 &= b.
 \end{aligned}$$



Podobne dokážeme aj druhú rovnosť využitím (3.4) a druhého riadku systému (3.3):

$$\begin{aligned}
 A^T y(\alpha) + s(\alpha) &= A^T (y + \alpha \Delta y) + s + \alpha \Delta s \\
 &= A^T y + c + \alpha (A \Delta y + \Delta s) \\
 &= c + \alpha 0 \\
 &= c.
 \end{aligned}$$

Pre overenie podmienky kladnosti  $(x(\alpha), s(\alpha)) > 0$  poznamenajme, že  $(x, s) = (x(0), s(0)) > 0$ .

Zo vzťahu (3.18) vyplýva:

$$x_i(\alpha) s_i(\alpha) \geq (1 - \theta) \sigma(\alpha) = (1 - \theta)(1 - \alpha(1 - \tau)) \sigma > 0, \quad (3.19)$$

kde ostrá nerovnosť je dôsledkom voľby parametrov  $\theta \in (0, 1)$ ,  $\tau \in (0, 1)$  a  $\alpha \in (0, 1]^5$ . Z toho vyplýva, že nemôžu nastať rovnosti  $x_i(\alpha) = 0$ , alebo  $s_i(\alpha) = 0$  pre akýkoľvek index  $i$  pri voľbe parametra  $\alpha \in [0, 1]$  a keďže sú  $x_i(\alpha)$  a  $s_i(\alpha)$  spojité funkcie v  $\alpha$ , tak nemôžu byť ani záporné. Preto musí platiť, že  $(x(\alpha), s(\alpha)) > 0$ ,  $\forall \alpha \in [0, 1]$  a teda bod  $(x(\alpha), y(\alpha), s(\alpha)) \in \mathcal{F}^o$ , čím sme dokázali platnosť vety.  $\square$

Pre kompletnosť dôkazu platnosti Algoritmu KK otestujme ešte, že aj špeciálna voľba parametrov  $\theta = 0,4$  a  $\tau = 1 - \frac{0,4}{\sqrt{n}}$  spĺňa podmienku (3.17) z Vety 3.3, t.j.

$$\frac{\theta^2 + n(1 - \tau)^2}{2^{3/2}(1 - \theta)} \leq \tau \theta \quad \forall n \geq 1.$$

Po dosadení hodnôt parametrov z Algoritmu KK do vzťahu dostávame:

$$\begin{aligned}
 \frac{(0,4)^2 + n(1 - 1 + \frac{0,4}{\sqrt{n}})^2}{2^{3/2}(1 - 0,4)} &\leq (1 - \frac{0,4}{\sqrt{n}})0,4 \\
 \frac{2(0,4)}{2^{3/2}(0,6)} &\leq (1 - \frac{0,4}{\sqrt{n}}) \\
 \frac{0,4}{\sqrt{n}} &\leq 1 - \frac{\sqrt{2}}{3} \\
 \sqrt{n} &\geq \frac{6}{5(3 - \sqrt{2})} \doteq 0,75 \\
 n &\geq 0,57
 \end{aligned}$$

z čoho jasne vidíme, že je nerovnosť (3.17) splnená pre všetky  $n \geq 1$ .

<sup>5</sup>Ak by bolo  $\alpha = 0$ , tak by sme sa nachádzali stále v tom istom bode, nespravili by sme žiaden krok.

### 3.2 Prediktor–korektor (PK) metódy

V Algoritme KK sme vybrali centrujúci parameter konštantný pre všetky iterácie  $\tau_k = \tau \in (0, 1)$ . Takáto voľba parametra má dosiahnuť v každom kroku, v každej iterácii dvojaký cieľ a to zlepšenie centrality a zredukovanie mierky duality. Na rozdiel od metód používajúcich takúto voľbu kroku, *prediktor–korektor metódy* (PK) a Algoritmus PK používajú na dosiahnutie uvedených cieľov dva rôzne druhy krokov, ktoré sa postupne striedajú v jednotlivých iteráciách. Sú to nasledovné dva druhy krokov:

- ◇ *prediktor krok* ( $\tau_k = 0$ ) slúži na redukcii parametra presnosti mierky duality  $\sigma$ . Pre tento krok dostávame, že  $\sigma(\alpha) = (1 - \alpha)\sigma$ .
- ◇ *korektor krok* ( $\tau_k = 1$ ) slúži na vylepšenie centrality. Keďže  $\sigma(\alpha) = \sigma$ , je zrejmé, že tento krok neovplyvňuje mierku presnosti duality.

Ďalšou dôležitou zložkou Algoritmu PK je dvojica okolí  $\mathcal{N}_2(\theta)$ , ktoré su umiestnené sústredne jedno v druhom. Budeme ich v ďalšom texte označovať ako  $\mathcal{N}_2(\theta_{in})$  – vnútorné okolie a  $\mathcal{N}_2(\theta_{out})$  – vonkajšie okolie.

Párne iterácie  $(x^k, y^k, s^k)$  (s párnym indexom  $k$ ) majú obmedzenie, aby sa nachádzali vo vnútornom okolí, zatiaľ čo nepárne iterácie majú o niečo väčšiu mieru voľnosti a postačí, ak sa budú nachádzať vo vonkajšom okolí – nie však mimo neho.

Budeme teraz skúmať prvé dve iterácie Algoritmu PK, čo postačí na ilustráciu celého algoritmu.

Štartujeme z bodu  $(x^o, y^o, s^o)$ , ktorý sa nachádza vo vútornom okolí. Rátame prediktor krok,  $\tau_o = 0$ . Pohybujeme sa pozdĺž tohoto smeru, až pokým sa nedostaneme nahranicu vonkajšieho okolia  $\mathcal{N}_2(\theta_{out})$  (nie však sa túto hranicu). V tomto bode zastaneme a priradíme hodnotu novej iterácii  $(x^1, y^1, s^1)$ . Ďalej pokračujeme rátaním korektor kroku,  $\tau_1 = 1$ . Dĺžku kroku v tomto smere volíme  $\alpha = 1$ , t.j. jednotkový krok. Ním sa presunieme do nového bodu, kde definujeme ďalšiu iteráciu  $(x^2, y^2, s^2)$ . Táto sa nachádza vo vnútornom okolí  $\mathcal{N}_2(\theta_{in})$ , čo neskor bližšie ukážeme v nasledujúcich tvrdeniach. Tento dvojkrokový cyklus opakujeme generujúc postupnosť iterácií, pričom párne iterácie sa nachádzajú vo vnútornom okolí a nepárne iterácie sa nachádzajú vo vonkajšom okolí. Skončíme pri dosiahnutí požadovanej presnosti riešenia.

Prediktor krok, ako je zrejmé z vyjadrenia  $\sigma(\alpha) = (1 - \alpha)\sigma$ , znižuje hodnotu parametra presnosti duality  $\sigma$  koeficientom  $(1 - \alpha)$ , pričom  $\alpha$  je koeficient dĺžky kroku. Korektor krok necháva  $\sigma$  nezmené, keďže  $\sigma(\alpha) = \sigma$ . Tým že sa cez korektor krok naspäť vraciame do vnútorného okolia, dávame tak v nasledujúcej iterácii s prediktor krokom väčší priestor pre manipuláciu.

Formálny zápis Algoritmu PK môžeme popísať vychádzajúc zo skeletu algoritmu z kapitoly 2.3.

### 3.2.1 Algoritmus PK

**Vstup:**  $(x^o, y^o, s^o) \in \mathcal{N}_2(\theta_{in})$

**pre**  $k = 0, 1, 2, \dots$

**Ak:**  $k$  je párne

*\*prediktor krok\**

1. rieš:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -X^k S^k e \end{bmatrix}, \quad (3.20)$$

2. polož:

$$\alpha_k = \max\{\alpha \mid (x^k(\alpha), y^k(\alpha), s^k(\alpha)) \in \mathcal{N}_2(\theta_{out}), \alpha \in [0, 1]\},$$

3. prirad:

$$(x^{k+1}, y^{k+1}, s^{k+1}) = (x^k(\alpha_k), y^k(\alpha_k), s^k(\alpha_k)),$$

inak:

*\*korektor krok\**

4. rieš:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -X^k S^k e + \sigma_k e \end{bmatrix}, \quad (3.21)$$

5. prirad:

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

skonči.

Väčšina analýz uskutočnených pre metódy s krátkym krokom má všeobecnú platnosť aj pre prediktor–korektor metódy, keďže v oboch prípadoch používame voľbu  $\mathcal{N}_2(\theta)$  okolia. Preto teraz budeme ďalej analyzovať špecifiká pre prediktor–korektor metódy, ako je voľba maximálneho možného  $\alpha_k$  v kroku 2. a taktiež vzťah medzi  $\theta_{out}$  a  $\theta_{in}$ , pri ktorom je zabezpečená funkčnosť Algoritmu PK. Pripomeňme si definíciu okolia aplikovanú pre voľbu parametrov  $\theta_{out}$  a  $\theta_{in}$ :

$$\mathcal{N}_2(\theta_{out}) = \left\{ (x, y, s) \in \mathcal{F}^o \mid \left\| \frac{XSe}{\sigma} - e \right\|_2 \leq \theta_{out} \right\},$$

$$\mathcal{N}_2(\theta_{in}) = \left\{ (x, y, s) \in \mathcal{F}^o \mid \left\| \frac{XSe}{\sigma} - e \right\|_2 \leq \theta_{in} \right\}.$$

Nasledujúca lema opisuje správanie sa každého prediktor kroku a zároveň určuje spodnú hranicu dĺžky prediktor kroku a dáva odhad na redukciu  $\sigma$ .

**Lema 3.5** *Predpokladajme, že  $(x, y, s) \in \mathcal{N}_2(\theta_{in})$  a  $(\Delta x, \Delta y, \Delta s)$  je definovaný systémom (3.20)<sup>6</sup>.*

*Potom  $(x(\alpha), y(\alpha), s(\alpha)) \in \mathcal{N}_2(\theta_{out})$ ,  $\forall \alpha \in [0, \bar{\alpha}]$ , kde*

$$\bar{\alpha} = \min \left( \frac{1}{2}, \left( \frac{\eta\sigma}{\|\Delta X \Delta S e\|} \right)^{1/2} \right) \quad (3.22)$$

a

$$\eta = \frac{\theta_{out} - \theta_{in}}{2}. \quad (3.23)$$

*Z toho vyplýva, že dĺžka kroku je prinajmenšom  $\bar{\alpha}$  a nová hodnota mierky duality  $\sigma$  dosahuje hodnotu najviac  $(1 - \bar{\alpha})\sigma$ .*

*Dôkaz:* Z Lemy (3.4) máme:

$$\begin{aligned} & \|X(\alpha)S(\alpha)e - \sigma(\alpha)e\| \\ & \leq |1 - \alpha| \|XSe - \sigma e\| + \alpha^2 \|\Delta X \Delta S e\| \\ & \leq (1 - \alpha) \|XSe - \sigma e\| + \frac{\eta\sigma}{\|\Delta X \Delta S e\|} \|\Delta X \Delta S e\| \quad \text{z (3.22)} \\ & \leq (1 - \alpha)\sigma\theta_{in} + \frac{\eta\sigma}{(1 - \alpha)}(1 - \alpha) \quad (x, y, s) \in \mathcal{N}_2(\theta_{in}) \\ & = (1 - \alpha)\sigma \left( \theta_{in} + \frac{\eta}{(1 - \alpha)} \right) \\ & \leq (1 - \alpha)\sigma(\theta_{in} + 2\eta) \quad \text{keďže } \alpha \leq \frac{1}{2} \\ & = \sigma(\alpha)(\theta_{in} + 2\eta) \quad \text{z (3.6) pre } \tau = 0 \\ & = \sigma(\alpha)(\theta_{out}) \quad \text{kde } \theta_{out} = \theta_{in} + 2\eta \end{aligned}$$

Odtiaľ dostávame platnosť tvrdenia, že  $(x(\alpha), y(\alpha), s(\alpha))$  patrí okoliu  $\mathcal{N}_2(\theta_{out})$  pre všetky hodnoty  $\alpha \in [0, \bar{\alpha}]$ , a to pri voľbe parametra  $\eta = \frac{\theta_{out} - \theta_{in}}{2}$ .

Pre úplnosť nám zostáva ešte overiť podmienku ostrej prípustnosti, čo dokážeme obdobne ako vo Vete 3.3:

Prvé dve podmienky

$$\begin{aligned} Ax(\alpha) &= b \\ A^T y(\alpha) + s(\alpha) &= c \end{aligned}$$

<sup>6</sup>Poznamenajme, že prediktor krok je definovaný systémom (2.14), pri voľbe parametra  $\tau = 0$ .

triviálne vyplývajú z (3.4) a z prvých dvoch riadkov systému (3.20):

$$Ax(\alpha) = A(x + \alpha\Delta x) = Ax + \alpha A\Delta x = b$$

$$A^T y(\alpha) + s(\alpha) = A^T(y + \alpha\Delta y) + s + \alpha\Delta s = c.$$

Overme ďalej platnosť podmienky kladnosti  $(x(\alpha), s(\alpha)) > 0$ .

Vychádzajúc zo vzťahu (3.18), pre  $\tau = 0$  dostávame:

$$x_i(\alpha)s_i(\alpha) \geq (1 - \theta_{in})\sigma(\alpha) = (1 - \theta_{in})(1 - \alpha)\sigma > 0,$$

$\forall i$  a  $\forall \alpha \in [0, 1]$ . Ostrá nerovnosť je dôsledkom voľby parametrov  $\theta_{in} \in (0, 1)$  a  $\alpha \in (0, \bar{\alpha}]$ . Pre  $\alpha = 0$  je ostrá nerovnosť splnená triviálne, keďže  $(x(0), s(0)) = (x, s) > 0$ . Vtedy máme  $x_i(0) > 0$  a  $s_i(0) > 0$ . Funkcie  $x_i(\alpha)$  a  $s_i(\alpha)$  sú spojité v  $\alpha$  a teda  $x_i(\alpha)$  ani  $s_i(\alpha)$  nemôže byť rovná nule pre žiadne  $\alpha \in [0, 1]$ , a teda nemôže byť ani záporné.

Týmto sme overili podmienku ostrej prípustnosti a teda aj platnosť lemy.  $\square$

Na určenie spodnej hranice  $\bar{\alpha}$  môže byť použitý odhad z Lemy 3.3. Pri voľbe parametra  $\tau = 1$  (prediktor krok) a pre každé  $\theta \in (0, 1)$  dostávame odhad:

$$\begin{aligned} \frac{\eta\sigma}{\|\Delta X \Delta S e\|} &\geq \frac{\eta\sigma 2^{3/2}(1-\theta)}{\sigma(\theta^2 + n(1-\tau)^2)} \\ &\geq \frac{\eta 2^{3/2}(1-\theta)}{\theta^2 + n} \\ &\geq \frac{\eta 2^{3/2}}{n} \quad \theta \in (0, 1) \\ &\geq \frac{\eta}{n}, \end{aligned}$$

pre všetky  $n \geq 1$ .

Na základe definície  $\bar{\alpha}$  z (3.22) dostávame odhad:

$$\bar{\alpha} \geq \min\left(\frac{1}{2}, \left(\frac{\eta}{n}\right)^{1/2}\right).$$

Pri prediktor kroku (teda pre párne iterácie) táto hranica implikuje redukciu parametra presnosti mierky duality:

$$\sigma_{k+1} \leq \left(1 - \left(\frac{\eta}{n}\right)^{1/2}\right) \sigma_k, \quad k = 0, 2, 4, \dots \quad (3.24)$$

Korektor kroky sú opísané v nasledujúcej leme. Pre korektor kroky ukážeme, pri akých hodnotách parametrov  $\theta_{out}$  a  $\theta_{in}$  vrátia akýkoľvek bod z okolia  $\mathcal{N}_2(\theta_{out})$  do vnútorného okolia  $\mathcal{N}_2(\theta_{in})$  a to bez dopadu na hodnotu parametra mierky presnosti duality  $\sigma$ .

**Lema 3.6** *Predpokladajme, že  $(x, y, s) \in \mathcal{N}_2(\theta_{out})$  a že  $(\Delta x, \Delta y, \Delta s)$  je definované systémom (3.21)<sup>7</sup>.*

*Potom platí, že:*

$$\sigma(1) = \sigma$$

a

$$(x(1), y(1), s(1)) \in \mathcal{N}_2(\theta_{in}) \quad \forall \theta_{in} \geq \frac{\theta_{out}^2}{2^{3/2}(1 - \theta_{out})}. \quad (3.25)$$

*Dôkaz:* Dosadením  $\tau = 1$  do vzťahu (3.6) dostávame:

$$\sigma(1) = (1 - \alpha(1 - 1))\sigma = \sigma \quad \forall \alpha \in [0, 1].$$

Dosadením  $\theta = \theta_{out}$ ,  $\alpha = 1$  a  $\tau = 1$  do tvrdenia z Lemy 3.4 dostávame:

$$\begin{aligned} \|X(1)S(1)e - \sigma(1)e\| &\leq |1 - 1|\theta_{out}\sigma + 1^2 \frac{\theta_{out}^2 + n(1 - 1)^2}{2^{3/2}(1 - \theta_{out})}\sigma \\ &= 0 + \frac{\theta_{out}^2}{2^{3/2}(1 - \theta_{out})}\sigma \\ &= \frac{\theta_{out}^2}{2^{3/2}(1 - \theta_{out})}\sigma(1) \\ &= \theta_{in}\sigma(1), \quad \text{kde } \theta_{in} = \frac{\theta_{out}^2}{2^{3/2}(1 - \theta_{out})}. \end{aligned}$$

Splnenie podmienky presnosti sme týmto dokázali. Pre úplnosť dôkazu je ešte potrebné, aby sme preverili podmienku ostrej prípustnosti pre bod  $(x(1), y(1), s(1))$ , čo by sme ukázali podobne ako v dôkaze Vety 3.3 a Lemy 3.5 overením všetkých požadovaných podmienok.  $\square$

Ak by sme v (3.25) zvolili  $\theta_{in}$  tak, aby platila rovnosť (teda najmenšie možné), vtedy pre najväčšie možné  $\theta_{out}$  dostávame nasledovné horné ohraničenie:

$$\begin{aligned} (0 <) \theta_{in} = \frac{\theta_{out}^2}{2^{3/2}(1 - \theta_{out})} &\leq \theta_{out} (< 1) \\ \frac{\theta_{out}}{2^{3/2}(1 - \theta_{out})} &\leq 1 \\ \theta_{out} &\leq 2^{3/2} - 2^{3/2}\theta_{out} \\ \theta_{out}(1 + 2^{3/2}) &\leq 2^{3/2} \\ \theta_{out} &\leq \frac{2^{3/2}}{1 + 2^{3/2}} \doteq 0,74 (< 1). \end{aligned}$$

<sup>7</sup>Poznamenajme, že korektor krok je definovaný systémom (2.14), pri voľbe parametra  $\tau = 1$ .

Všimnime si, že hodnoty  $\theta_{in} = 0,25$  a  $\theta_{out} = 0,5$  spĺňajú nerovnosť (3.25):

$$\begin{aligned} 0,25 &\geq \frac{(0,5)^2}{2^{3/2}(1-0,5)} \\ 0,25 &\geq \frac{(0,25)}{2^{3/2}0,5} \\ 1 &\geq \frac{1}{\sqrt{2}}. \end{aligned}$$

Ako vidíme zo znenia lemy, iterácie korektor kroku nechávajú mierku duality  $\sigma$  nezmenenú. Keďže prediktor iterácie dosahujú významnú redukciu  $\sigma$  (viď vzťah (3.24)), možno preto dokázať pre Algoritmus PK ten istý druh polynomiálnej zložitosti, ako pri Algoritme KK.

**Veta 3.4** *Nech  $\epsilon \in (0,1)$ . Predpokladajme, že štartujúci bod  $(x^o, y^o, s^o) \in \mathcal{N}_2(\theta_{in})$  v Algoritme PK vyhovuje podmienke:*

$$\sigma_o \leq \frac{1}{\epsilon^\kappa} \quad (3.26)$$

pre nejakú pozitívnu konštantu  $\kappa$ .

Potom existuje index  $K$ :

$$K = O(\sqrt{n} \log \frac{1}{\epsilon}),$$

taký, že

$$\sigma_k \leq \epsilon \quad \forall k \geq K.$$

*Dôkaz:* Kombináciou (3.24) a predchádzajúcej Lemy 3.6 dostávame:

$$\sigma_{k+2} = \sigma_{k+1} \leq \left(1 - \left(\frac{\eta}{n}\right)^{1/2}\right) \sigma_k \quad k = 0, 2, 4, \dots \quad (3.27)$$

Z toho vyplýva, že požiadavka redukcie (3.9) z Vety 3.1 je splnená, keď položíme  $\delta = \eta^{1/2}$  a  $\omega = \frac{1}{2}$  pre každú druhú iteráciu. To znamená, že vo všeobecnosti máme vzťah:

$$\sigma_{k+2} = \sigma_{k+1} \leq \left(1 - \frac{\delta}{n^\omega}\right) \sigma_k \quad k = 0, 2, 4, \dots, \quad (3.28)$$

ktorý však neovplyvní platnosť výsledkov Vety 3.1. Aplikujúc postup z dôkazu Vety 3.1 na vzťah (3.28) určíme horný odhad  $K$  na počet párnych iterácií, zabezpečujúci aby  $\sigma_k \leq \epsilon$ , pre každé  $k$  párne, také že  $k \geq K$ .

Vychádzajúc z (3.28) pre párne iterácie dostávame:

$$\sigma_k \leq \left(1 - \frac{\delta}{n^\omega}\right)^{\frac{k}{2}} \sigma_o$$

Zlogaritmovaním a úpravami dostávame:

$$\begin{aligned}\log \sigma_k &\leq \frac{k}{2} \log \left( 1 - \frac{\delta}{n^\omega} \right) + \log \sigma_0 \\ \log \sigma_k &\leq \frac{k}{2} \left( -\frac{\delta}{n^\omega} \right) + \kappa \log \frac{1}{\epsilon}.\end{aligned}$$

Podmienka  $\sigma_k \leq \epsilon$  je splnená, ak máme:

$$\begin{aligned}\frac{k}{2} \left( -\frac{\delta}{n^\omega} \right) + \kappa \log \frac{1}{\epsilon} &\leq \log \epsilon \\ \frac{k}{2} \left( -\frac{\delta}{n^\omega} \right) &\leq -\kappa \log \frac{1}{\epsilon} - \log \frac{1}{\epsilon} \\ \frac{k}{2} &\geq (1 + \kappa) \frac{n^\omega}{\delta} \log \frac{1}{\epsilon} \\ k &\geq K = 2(1 + \kappa) \frac{n^\omega}{\delta} \log \frac{1}{\epsilon}.\end{aligned}$$

Pretože pri neprárnych iteráciách je  $\sigma_k = \sigma_{k-1}$ , dostaneme, že  $\sigma_k \leq \epsilon$ , pre všetky  $k$ . Týmto je veta dokázaná.  $\square$

Algoritmus PK je značným vylepšením Algoritmu KK kvôli adaptívnej voľbe dĺžky prediktor kroku. V Algoritme KK sú hodnoty  $\tau$  a  $\alpha$  volené pevne, tak aby obmedzili iterácie  $(x^k, y^k, s^k)$  iba na okolie  $\mathcal{N}_2(\theta)$  a to za akýchkoľvek podmienok. Na porovnanie, dĺžka prediktor kroku z Algoritmu PK je dlhšia, ak je smer prediktora dobrým smerom, t.j. keď produkuje výraznejšie  $\sigma$  bez toho, aby sme sa dostali príliš ďaleko od centrálnej trajektórie  $\mathcal{C}$ . V posledných krokoch Algoritmu PK sa smerovanie prediktor kroku vylepšuje natoľko, že sme schopní použiť dĺžku kroku  $\alpha$  blízko 1.

Aj napriek adaptívnej voľbe kroku  $\alpha$  je Algoritmus PK stále dosť obmedzujúci kvôli svojej požiadavke, aby sa iterácie nachádzali v okolí  $\mathcal{N}_2(\theta)$ . Táto požiadavka je veľmi obmedzujúca najmä počas prvých iterácií, keď sa nachádzame ďaleko od optimálneho riešenia.

V ďalšej časti opíšeme metódy s dlhým krokom (DK), ktoré kombinujú flexibilitu voľby dĺžky kroku s voľbou liberálnejšieho okolia  $\mathcal{N}_{-\infty}(\gamma)$ .

### 3.3 Metódy s dlhým krokom (DK)

Algoritmus používaný pre metódy s dlhým krokom generuje postupnosť iterácií nachádzajúcich sa v okolí  $\mathcal{N}_{-\infty}(\gamma)$ , pričom pre veľmi malé hodnoty parametra  $\gamma$ ,



blízke nule, pokrýva okolie takmer celý priestor ostro prípustných bodov  $\mathcal{F}^o$ . Na štartujúci bod je rovnako kladená požiadavka, aby sa nachádzal v okolí  $\mathcal{N}_{-\infty}(\gamma)$ . V každej iterácii algoritmu vyberáme centrujúci parameter  $\tau_k$  tak, aby ležal medzi dvoma fixnými hranicami  $\tau_{min}$  a  $\tau_{max}$ , pričom platí, že  $0 < \tau_{min} < \tau_{max} < 1$ . Smer jednotlivých iterácií dostaneme riešením systému (2.14), dĺžku kroku  $\alpha$  volíme podľa možností čo najväčšiu s ohľadom na to, aby sa všetky iterácie nachádzali v okolí  $\mathcal{N}_{-\infty}(\gamma)$ . Opäť na základe skeletu algoritmu z kapitoly 2.3, strana 12 vyjadríme kostru algoritmu pre metódy s dlhým krokom.

### 3.3.1 Algoritmus DK

**Vstup:**  $\gamma \in (0, 1)$

$$\begin{aligned} & \tau_{min}, \tau_{max} \quad 0 < \tau_{min} < \tau_{max} < 1 \\ & (x^o, y^o, s^o) \in \mathcal{N}_{-\infty}(\gamma) \end{aligned}$$

pre  $k = 0, 1, 2, \dots$

**1. vyber:**  $\tau_k \in (\tau_{min}, \tau_{max})$

**2. rieš:**

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -X^k S^k e + \tau_k \sigma_k e \end{bmatrix}, \quad (3.29)$$

**3. polož:**  $\alpha_k = \max\{\alpha \mid (x(\alpha), y(\alpha), s(\alpha)) \in \mathcal{N}_{-\infty}(\gamma), \alpha \in [0, 1]\}$ ,

**4. prirad:**

$$(x^{k+1}, y^{k+1}, s^{k+1}) = (x^k(\alpha_k), y^k(\alpha_k), s^k(\alpha_k)),$$

**skonči.**

Dolná hranica centrujúceho parametra  $\tau_{min}$  zabezpečí, že generované smery iterácií sa budú pohybovať od hranice okolia  $\mathcal{N}_{-\infty}(\gamma)$  smerom do vnútra okolia. Je to smer vylepšujúci centralitu, ktorý je efektívny pre malé hodnoty parametra dĺžky kroku  $\alpha$ . Veľké hodnoty  $\alpha$ , blízke jednotke, nás dostávajú späť za hranicu okolia.<sup>8</sup> Znenie nasledujúcich viet charakterizuje minimálnu dĺžku kroku  $\alpha$ , závislého od vstupných parametrov, pri ktorej máme zaručené, že neopustíme okolie  $\mathcal{N}_{-\infty}(\gamma)$ . Súčasne popíšeme odhad chyby aproximácie a skonštatujeme polynomiálnu zložitosť.

<sup>8</sup>Chyba aproximácie nelineárneho systému lineárnym systémom sa zväčšuje s rastúcim  $\alpha$ .

**Lema 3.7** Ak  $(x, y, s) \in \mathcal{N}_{-\infty}(\gamma)$ , potom

$$\|\Delta X \Delta S e\| \leq 2^{-3/2} \left(1 + \frac{1}{\gamma}\right) n\sigma.$$

*Dôkaz:* Podobne ako v dôkaze Lemy 3.3 budeme vychádzať z nerovnosti:

$$\|\Delta X \Delta S e\| \leq 2^{-3/2} \|(XS)^{-1/2}(-XSe + \tau\sigma e)\|^2$$

Umocnením normy a nahradením vzťahov  $x^T s = n\sigma$  a  $e^T e = n$  v predchádzajúcej nerovnosti dostávame:

$$\begin{aligned} \|\Delta X \Delta S e\| &\leq 2^{-3/2} \left\| -(XS)^{1/2} e + \tau\sigma (XS)^{-1/2} e \right\|^2 \\ &\leq 2^{-3/2} \left[ x^T s - 2\tau\sigma e^T e + \tau^2 \sigma^2 \sum_{i=1}^n \frac{1}{x_i s_i} \right] \\ &\leq 2^{-3/2} \left[ x^T s - 2\tau\sigma n + \tau^2 \sigma^2 \frac{n}{\gamma\sigma} \right] \quad \text{keďže } x_i s_i \geq \gamma\sigma \\ &\leq 2^{-3/2} \left[ n\sigma - 2\tau n\sigma + \frac{\tau^2}{\gamma} n\sigma \right] \\ &= 2^{-3/2} \left[ 1 - 2\tau + \frac{\tau^2}{\gamma} \right] n\sigma \\ &\leq 2^{-3/2} \left[ 1 + \frac{1}{\gamma} \right] n\sigma \quad \text{keďže } \tau \in (0, 1), \end{aligned}$$

čím sme dokázali platnosť lemy pre odhad chyby aproximácie.  $\square$

Ako vidíme, Lema 3.7 je podobným výsledkom pre metódy s dlhým kormom, akým bola Lema 3.3 pre metódy s krátkym krokom. Lemou 3.3 sú učené užšie hranice, keďže na aktuálny bod  $(x, y, s)$  kladieme požiadavku, aby ležal v reštriktívnejšom okolí cetrálnej trajektórie  $\mathcal{N}_2(\theta)$ . Rozdielanosti v hraniciach, ktoré dostávame pre  $\Delta X \Delta S e$  vychádzajú z rozdielov polynomiálnej zložitosti algoritmov s krátkym krokom a s dlhým krokom.

**Veta 3.5** Pri daných parametroch  $\gamma$ ,  $\tau_{\min}$  a  $\tau_{\max}$  v Algoritme DK existuje konštanta  $\delta$ , nezávislá od  $n$ , taká že:

$$\sigma_{k+1} \leq \left(1 - \frac{\delta}{n}\right) \sigma_k \quad \forall k \geq 0.$$

*Dôkaz:* Na začiatok dokážeme, že:

$$(x^k(\alpha), y^k(\alpha), s^k(\alpha)) \in \mathcal{N}_{-\infty}(\gamma) \quad \forall \alpha \in \left[0, 2^{3/2} \gamma \frac{1 - \gamma \tau_k}{1 + \gamma n}\right], \quad (3.30)$$

odkiaľ je zrejmé, že  $\alpha_k$  je ohraničená:

$$\alpha_k \geq 2^{3/2} \gamma \frac{1 - \gamma \tau_k}{1 + \gamma n}. \quad (3.31)$$

Z predchádzajúcej Lemy 3.7 dostávame, že pre všetky  $i = 1, 2, \dots, n$  platí:

$$|\Delta x_i^k \Delta s_i^k| \leq \|\Delta X \Delta S e\| \leq 2^{-3/2} \left(1 + \frac{1}{\gamma}\right) n \sigma. \quad (3.32)$$

Na základe skutočnosti, že sa nachádzame v okolí  $\mathcal{N}_{-\infty}(\gamma)$  (t.j. platí  $x_i s_i \geq \gamma \sigma_k$ ), použitím tretieho riadku systému (3.29), ktorým je definovaný smer iterácií  $(\Delta x^k, \Delta y^k, \Delta s^k)$  a využitím predchádzajúceho vzťahu (3.32) dostávame:

$$\begin{aligned} x_i^k(\alpha) s_i^k(\alpha) &= (x_i^k + \alpha \Delta x_i^k)(s_i^k + \alpha \Delta s_i^k) \\ &= x_i^k s_i^k + \alpha(x_i^k \Delta s_i^k + s_i^k \Delta x_i^k) + \alpha^2 \Delta x_i^k \Delta s_i^k \\ &= x_i^k s_i^k + \alpha(-x_i^k s_i^k + \tau_k \sigma_k) + \alpha^2 \Delta x_i^k \Delta s_i^k \\ &\geq x_i^k s_i^k (1 - \alpha) + \alpha \tau_k \sigma_k - \alpha^2 |\Delta x_i^k \Delta s_i^k| \\ &\geq \gamma(1 - \alpha) \sigma_k + \alpha \tau_k \sigma_k - \alpha^2 2^{-3/2} \left(1 + \frac{1}{\gamma}\right) n \sigma_k \end{aligned}$$

Zo vzťahu (3.6) pre  $k$ -tu iteráciu platí:

$$\sigma_k(\alpha) = (1 - \alpha(1 - \tau_k)) \sigma_k.$$

Na základe posledných dvoch vzťahov môžeme vidieť, že podmienka presnosti

$$x_i^k(\alpha) s_i^k(\alpha) \geq \gamma \sigma_k(\alpha) \quad (3.33)$$

je splnená ak platí:

$$\gamma(1 - \alpha) \sigma_k + \alpha \tau_k \sigma_k - \alpha^2 2^{-3/2} \left(1 + \frac{1}{\gamma}\right) n \sigma_k \geq \gamma(1 - \alpha(1 - \tau_k)) \sigma_k,$$

z čoho ďalej po úpravách dostávame:

$$\begin{aligned} \gamma \sigma_k - \gamma \alpha \sigma_k + \alpha \tau_k \sigma_k - \alpha^2 2^{-3/2} \left(1 + \frac{1}{\gamma}\right) n \sigma_k &\geq \gamma \sigma_k - \gamma \alpha \sigma_k + \gamma \alpha \tau_k \sigma_k \\ \alpha(1 - \gamma) \tau_k \sigma_k &\geq \alpha^2 2^{-3/2} \left(1 + \frac{1}{\gamma}\right) n \sigma_k, \end{aligned}$$

čo platí, ak:

$$\alpha \leq 2^{3/2} \gamma \frac{1 - \gamma \tau_k}{1 + \gamma n}.$$

Týmto sme overili, že bod  $(x^k(\alpha), y^k(\alpha), s^k(\alpha))$  spĺňa podmienku presnosti pre okolie  $\mathcal{N}_{-\infty}(\gamma)$ , ak:

$$\alpha \in \left[ 0, 2^{3/2} \gamma \frac{1 - \gamma \tau_k}{1 + \gamma n} \right].$$

Ďalej dokážeme, že  $(x^k(\alpha), y^k(\alpha), s^k(\alpha)) \in \mathcal{F}^o$  pre všetky  $\alpha$  z daného intervalu. Dôkaz prevedieme analogicky ako vo vete 3.3.

Overíme všetky podmienky kladené na bod patriaci množine  $\mathcal{F}^o$ .

Splnenie prvých dvoch podmienok

$$\begin{aligned} Ax(\alpha) &= b \\ A^T y(\alpha) + s(\alpha) &= c \end{aligned}$$

triviálne vypláva z (3.4) a z prvých dvoch riadkov systému (3.29):

$$Ax(\alpha) = A(x + \alpha \Delta x) = Ax + \alpha A \Delta x = b$$

$$A^T y(\alpha) + s(\alpha) = A^T (y + \alpha \Delta y) + s + \alpha \Delta s = c.$$

Overme ešte podmienku kladnosti  $(x(\alpha), s(\alpha)) > 0$ . Keďže  $(x, s) > 0$ , pre  $\alpha = 0$  dostávame, že  $(x(0), s(0)) = (x, s) > 0$ .

Z predchádzajúcich úvah dôkazu a vychádzajúc z (3.33) dostávame:

$$x_i(\alpha) s_i(\alpha) \geq \gamma \sigma(\alpha) = \gamma(1 - \alpha(1 - \tau)) \sigma > 0. \quad (3.34)$$

Ostá nerovnosť je zabezpečená voľbou parametrov, pre ktoré platí, že  $\gamma \in (0, 1)$ ,  $\tau \in (0, 1)$ ,  $\alpha \in \left( 0, 2^{3/2} \gamma \frac{1 - \gamma \tau_k}{1 + \gamma n} \right]$  a spojitou funkciou  $x_i(\alpha)$  a  $s_i(\alpha)$ , z ktorej vyplýva, že  $x_i(\alpha) > 0$  a  $s_i(\alpha) > 0$ . Pre  $\alpha = 0$  je ostrá nerovnosť splnená triviálne. Týmto sme dokázali splnenie podmienky kladnosti a teda aj platnosť (3.30) a následne (3.31).

Dôkaz bude kompletný odhadnutím redukcie parametra  $\sigma$  v  $k$ -tom kroku. Zo vzťahov (3.6) a (3.31) dostávame:

$$\begin{aligned} \sigma_{k+1} &= (1 - \alpha_k(1 - \tau_k)) \sigma_k \\ &\leq \left( 1 - 2^{3/2} \gamma \frac{1 - \gamma \tau_k}{1 + \gamma n} (1 - \tau_k) \right) \sigma_k. \end{aligned} \quad (3.35)$$

Funkcia  $\tau(1 - \tau)$  je kvadraticá konkávna funkcia premennej  $\tau$  a teda na ľubovoľnom intervale z jej definičného oboru dosiahne svoje minimum v niektorom z krajných bodov intervalu. Z tohoto vyplýva, že máme:

$$\tau_k(1 - \tau_k) \geq \min\{\tau_{\min}(1 - \tau_{\min}); \tau_{\max}(1 - \tau_{\max})\} \quad \forall \tau_k \in (\tau_{\min}; \tau_{\max}).$$

Po dosadení odhadu do (3.35) dostávame:

$$\begin{aligned}\sigma_{k+1} &\leq \left(1 - 2^{3/2} \frac{\gamma}{n} \frac{1-\gamma}{1+\gamma} \tau_k (1-\tau_k)\right) \sigma_k \\ &\leq \left(1 - 2^{3/2} \frac{\gamma}{n} \frac{1-\gamma}{1+\gamma} \min\{\tau_{\min}(1-\tau_{\min}); \tau_{\max}(1-\tau_{\max})\}\right) \sigma_k\end{aligned}$$

z čoho pre

$$\delta = 2^{3/2} \frac{\gamma}{n} \frac{1-\gamma}{1+\gamma} \min\{\tau_{\min}(1-\tau_{\min}); \tau_{\max}(1-\tau_{\max})\}$$

dostávame platnosť tvrdenia tejto vety

$$\sigma_{k+1} \leq \left(1 - \frac{\delta}{n}\right) \sigma_k \quad \forall k \geq 0. \quad \square$$

Bezprostredným dôsledkom predchádzajúcej vety 3.5 a vety 3.1 je výsledok zložitosti Algoritmu DK.

**Veta 3.6** *Nech  $\epsilon > 0$  a  $\gamma \in (0, 1)$ . Predpokladajme, že pre štartujúci bod  $(x^o, y^o, s^o) \in \mathcal{N}_{-\infty}(\gamma)$  v Algoritme DK platí:*

$$\sigma_o \leq \frac{1}{\epsilon^\kappa}$$

pre nejaké  $\kappa > 0$ .

Potom existuje index  $K$ , kde  $K = O(n \log \frac{1}{\epsilon})$ , taký že:

$$\sigma_k \leq \epsilon \quad \forall k \geq K.$$

# Kapitola 4

## Mehrotrov algoritmus

Väčšina zdrojových kódov programov pre metódy vnútorného bodu je založená na Mehrotrovom prediktor–korektor algoritme (MPK). Mehrotrov algoritmus sa štruktúrou mierne odlišuje od Skeletu algoritmu zo strany 12 v tom zmysle, že klasický Newtonov smer je tu vylepšený o korekčnú zložku, ktorá je z výpočtového hľadiska nenáročná, keďže v nej budeme využívať informáciu získanú z riešenia systému pre hľadanie afínno–škálovacieho smeru. Algoritmus taktiež povoľuje adaptívnu voľbu centrujúceho parametra v každej iterácii.<sup>1</sup> Mehrotrov algoritmus sa vyznačuje vysokým stupňom aproximácie k centrálnej trajektórii. Ďalšou črtou tohoto algoritmu je, že pripúšťa aj neprípustné vnútorné body. Mehrotrova správnym smerom skombinoval doterajšie poznatky s dômyselnou heuristikou pre adaptívnu voľbu centrujúceho parametra  $\tau$ , dĺžky kroku a voľby štartujúceho bodu  $(x^o, y^o, s^o)$ . Výsledkom tejto jeho snahy je vysoko efektívny algoritmus, ktorý je, ako sme už spomenuli, základom väčšiny zdrojových kódov programov pre metódy vnútorného bodu.

### 4.1 Neprípustné štartujúce body

Doposiaľ sme uvažovali, že štartujúci bod  $(x^o, y^o, s^o)$  je ostro prípustný a spĺňa lineárne rovnosti  $Ax^o = b$  a  $A^T y^o + s^o = c$ . Všetky nasledujúce iterácie taktiež vyhovujú týmto rovnostiam, čo je zabezpečené nulovou pravou stranou v systéme (2.14) zo Skeletu algoritmu. Pre väčšinu úloh lineárneho programovania je náročné splnenie podmienky ostrej prípustnosti štartujúceho bodu. Tento problém môžeme celkom ľahko zmeniť preformulovaním úlohy. Takáto úprava môže mať však za následok to, že zväčšíme náročnosť riešenia úlohy.

Alternatívny prístup k riešeniu takýchto úloh ponúkajú *metódy neprípustných bodov*, v ktorých požadujeme od štartujúceho bodu iba splnenie podmienky klad-

---

<sup>1</sup>V Prediktor–korektor algoritme rátame centrujúci smer v každej druhej iterácii.

nosti  $(x^o, s^o) > 0$ .<sup>2</sup> Hľadaný smer vyžaduje, aby bol tak modifikovaný, nech smeruje bližšie k prípustným riešeniam a zároveň aj bližšie k centrálnej trajektórii  $\mathcal{C}$ . Docielime to malou zmenou v systéme (2.14). Pre túto potrebu si definujeme nasledovným spôsobom reziduá:

$$\begin{aligned} r_b &= Ax - b \\ r_c &= A^T y + s - c \end{aligned} \quad (4.1)$$

Smer hľadaný metódou neprípustných bodov dostávame riešením systému:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_b \\ -r_c \\ -XSe + \tau\sigma e \end{bmatrix}. \quad (4.2)$$

Vyrátany smer  $(\Delta x, \Delta y, \Delta s)$  stále predstavuje Newtonov krok k bodu  $(x_{\tau\sigma}, y_{\tau\sigma}, s_{\tau\sigma})$  patriacemu centrálnej trajektórii  $\mathcal{C}$ . Modifikáciou systému pomocou reziduí je tento smer ale tlačný, aby eliminoval neprípustnosť, a to pokiaľ možno hneď v prvom kroku. V takom prípade je zvolený plný krok (t.j.  $\alpha = 1$ ), reziduá sa stanú nulové a všetky nasledujúce iterácie už spĺňajú podmienku ostrej prípustnosti. Zväčša sa to však nepodarí docieľiť hneď v prvom kroku.

## 4.2 Charakteristika Mehrotrovho algoritmu

Mehrotrov algoritmus generuje postupnosť neprípustných iterácií  $(x^k, y^k, s^k)$ , pre ktoré platí podmienka kladnosti  $(x^k, s^k) > 0$ . Hľadaný smer v každej iterácii pozostáva z troch zložiek:

- ◇ *afínno-škálovací "prediktor" smer* – je to čistý Newtonov smer pre funkciu  $F(x, y, s)$  definovanú (2.5).
- ◇ *centrujúci člen* – jeho hodnota je regulovaná adaptívnou voľbou centrujúceho parametra  $\tau$ .
- ◇ *"korektor" smer* – pokúša sa o kompenzáciu nelinearity z afínno-škálovacieho smeru.

Prvé dve zložky – afínno škálovací smer a centrujúci člen – svojou kombináciou vylepšujú štandardný krok zo systému (4.2), ktorý vychádza z neprípustného vnútorného bodu.

V Mehrotrovom algoritme sú rátané afínno-škálovacia zložka kroku a centrujúca zložka kroku, každá samostatne. Takouto stratégiou výpočtu dosiahneme

<sup>2</sup>Takýto prístup využíva Mehrotrov algoritmus.

podstatnú výhodu v tom, že môžeme adaptívne vyberať centrujúci parameter  $\tau$ , než ho zvoliť konštantne pre celý algoritmus.<sup>3</sup> Ak afínno-škálovací smer zabezpečí dobrý progres v redukcii mierky duality  $\sigma$ , pri dodržaní podmienky, že ostaneme vo vnútri kladného ortantu (t.j. platí  $(x, s) > 0$ ), tak v takejto iterácii je potrebné malé centrovanie a preto priradíme centrujúcemu parametru  $\tau$  malú hodnotu, blízku 0. Ak ale kvôli splneniu podmienky  $(x, s) > 0$  môžeme spraviť iba malý krok v afínno-škálovacom smere, je potrebné markantnejšie centrovanie a teda  $\tau$  volíme blízke 1.

Ako dôležitý nedostatok pri takomto druhu samostatného výpočtu centrujúceho parametra sa javí to, že potrebujeme riešiť v každej iterácii dva lineárne systémy namiesto jedného. Úsilie, ktoré na to potrebujeme vynaložiť nie je až také veľké, keďže oba systémy majú zhodnú maticu koeficientov. Preto nám stačí iba raz zrátať a upraviť maticu (do trojuholníkového tvaru) a potom následne previesť dva krát spätnú substitúciu, pre každú z dvoch rôznych pravých strán.

Afínno-škálovací smer dostaneme z lineárnej aproximácie rovností KKT podmienok (2.1)–(2.3). Je to čistý Newtonov smer pre funkciu  $F(x, y, s)$  definovanú v (2.5). Keďže je tento smer rátaný samostatne, môže byť ďalej použitý na ohodnotenie chyby lineárnej aproximácie. Tento poznatok nám pomôže zrátať "korekčný" krok, ktorý tvorí tretiu zložku Mehrotrovho smeru, a tak vylepšiť náš lineárny model (prvého stupňa) na kvadratický model (druhého stupňa).

Keďže centrujúca a korekčná zložka sú výsledkom riešenia lineárneho systému s tou istou maticou koeficientov, ako pre afínno-škálovací smer, a keďže sú od seba nezávislé, nie je žiaden dôvod na to, aby sme ich rátali každú samostatne. Môžeme ich jednoducho zlúčiť do jedného smeru a to sčítaním príslušných pravých strán rátanej iterácie a takto získame kombinovaný smer iba jednou spätnou substitúciou.

Ako protihodnotu nákladov vynaložených na úsilie rátania dvoch spätných substitúcií nám ponúka Mehrotrov algoritmus adaptívnu voľbu  $\tau$  a vylepšenie afínno-škálovacieho smeru v podobe vyšších rádov. V globále sú tieto náklady vynahradené značným znížením počtu celkových potrebných iterácií pre dosiahnutie zvolenej presnosti.

Existuje viacero modifikácií Mehrotrovho algoritmu, pričom všetky sú podobné, keďže vychádzajú z toho istého základu. V mehrotrovom algoritme, ktorý tu bude prezentovaný, budeme na rozdiel od predchádzajúcich algoritmov rátať rozdielne dĺžky krokov pre primárne a pre duálne premenné.

<sup>3</sup>Poznamenajme, že konštantné  $\tau$  volíme napríklad v Metódach s krátkym krokom, alebo i v Prediktor-korektor metódach, kde tomuto parametru striedavo priradzujeme hodnoty 0 a 1.



### 4.3 Popis Mehrotrovho algoritmu

Najskôr popíšeme všeobecne jednu iteráciu z Merhotrovho algoritmu a potom následne v ďalšej časti definujeme celý algoritmus. Kvôli lepšej prehľadnosti v tejto časti vynecháme pri popisovaní  $k$ -tej iterácie označenie  $k$  pre matice a pre vektory.

Majme bod  $(x, y, s)$ , taký že  $(x, s) > 0$ . *Afínno-škálovací smer*  $(\Delta x^{af}, \Delta y^{af}, \Delta s^{af})$  je daný riešením systému:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x^{af} \\ \Delta y^{af} \\ \Delta s^{af} \end{bmatrix} = \begin{bmatrix} -r_b \\ -r_c \\ -XSe \end{bmatrix}, \quad (4.3)$$

kde  $r_b$  a  $r_c$  sú reziduá definované v (4.1). Tento systém dostávame ak v (4.2) položíme  $\tau = 0$ , teda nerobíme žiadne centrovanie. Teraz hľadáme maximálnu možnú dĺžku kroku v tomto smere, až po hranicu kladného ortantu, pričom ju budeme hľadať samostatne pre primárne premenné a samostatne pre duálne premenné:

$$\begin{aligned} \alpha_{af}^P &= \max\{\alpha \in [0, 1] \mid x + \alpha \Delta x^{af} \geq 0\}, \\ \alpha_{af}^D &= \max\{\alpha \in [0, 1] \mid s + \alpha \Delta s^{af} \geq 0\}. \end{aligned}$$

Ako mierku efektívnosti afínno-škálovacieho smeru definujeme  $\sigma_{af}$  ako hypotetickú hodnotu  $\sigma$ , vychádzajúcu z plného kroku ku hranici:

$$\sigma_{af} = \frac{(x + \alpha_{af}^P \Delta x^{af})^T (s + \alpha_{af}^D \Delta s^{af})}{n}.$$

Ak je hodnota  $\sigma_{af} \ll \sigma$ , tak je afínno-škálovací smer dobrým smerom, ktorý zabezpečuje značný progres v redukcii hodnoty  $\sigma$ , a preto nám bude stačiť zvoliť centrujúci parameter  $\tau$  blízky nule. Na druhej strane, ak je hodnota  $\sigma_{af}$  len o málo menšia ako hodnota  $\sigma$ , tak potom vyberáme  $\tau$  blízke jednotke. Takáto voľba  $\tau$ , blízka jednotke, má za následok posunutie smerom bližšie k centrálnej trajektórii, do pozície, z ktorej môžeme v nasledujúcej iterácii dosiahnuť väčšiu redukciiu  $\sigma$ .

Mehrotra navrhol nasledujúcu heuristiku pre voľbu vhodného  $\tau$ , ktorá sa dôkladným testovaním vo výpočtoch ukázala byť efektívna:

$$\tau = \left( \frac{\sigma_{af}}{\sigma} \right)^3.$$

Na výpočet *centrujúcej zložky kroku*  $(\Delta x^{cn}, \Delta y^{cn}, \Delta s^{cn})$  potrebujeme riešiť lineárny systém s tou istou maticou koeficientov ako v systéme (4.3), pričom pravá strana bude nahradená vektorom  $(0, 0, \tau \sigma e)^T$ . Ako už bolo spomenuté, je efektívne rátať centrujúcu zložku spolu s korekčnou zložkou kroku, ktorú teraz popíšeme.

Pozrime sa teraz bližšie, ako je ovplyvnená  $i$ -tá zložka súčiny  $x_i s_i$ , z mierky presnosti duality, plným krokom z afínno-škálovacieho smeru. Vychádzajúc z (4.3) potom dostávame:

$$\begin{aligned} (x_i + \Delta x_i^{af})(s_i + \Delta s_i^{af}) &= x_i s_i + x_i \Delta s_i^{af} + s_i \Delta x_i^{af} + \Delta x_i^{af} \Delta s_i^{af} \\ &= \Delta x_i^{af} \Delta s_i^{af}. \end{aligned} \quad (4.4)$$

Z toho vidíme, že pri plnom kroku dostávame  $x_i s_i = \Delta x_i^{af} \Delta s_i^{af}$  namiesto očakávanej rovnosti nule.

Korekčná zložka kroku  $(\Delta x^{kr}, \Delta y^{kr}, \Delta s^{kr})$  má kompenzovať takúto odchylku od linearity, tým že modifikuje hľadaný smer, aby sa zložky  $x_i s_i$  približovali k nulovej hodnote a teda aby sme dostali lepšie hodnoty pre mierku presnosti duality.

Túto zložku kroku dostaneme riešením systému:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x^{kr} \\ \Delta y^{kr} \\ \Delta s^{kr} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -\Delta X^{af} \Delta S^{af} e \end{bmatrix}, \quad (4.5)$$

kde

$$\begin{aligned} \Delta X^{af} &= \text{diag}(\Delta x_1^{af}, \Delta x_2^{af}, \dots, \Delta x_n^{af}), \\ \Delta S^{af} &= \text{diag}(\Delta s_1^{af}, \Delta s_2^{af}, \dots, \Delta s_n^{af}). \end{aligned}$$

Na ohodnotenie korekčnej zložky sa teraz pozrime na dvojice, ktoré sú výsledkom plného kroku v kombinovanom smere afínno-škálovacej zložky a korekčnej zložky. Z (4.4) a (4.5) máme:

$$\begin{aligned} (x_i + \Delta x_i^{af} + \Delta x_i^{kr})(s_i + \Delta s_i^{af} + \Delta s_i^{kr}) &= x_i s_i + x_i \Delta s_i^{af} + x_i \Delta s_i^{kr} \\ &\quad + s_i \Delta x_i^{af} + \Delta x_i^{af} \Delta s_i^{af} + \Delta x_i^{af} \Delta s_i^{kr} \\ &\quad + s_i \Delta x_i^{kr} + \Delta x_i^{kr} \Delta s_i^{af} + \Delta x_i^{kr} \Delta s_i^{kr} \\ &= \Delta x_i^{af} \Delta s_i^{kr} + \Delta x_i^{kr} \Delta s_i^{af} + \Delta x_i^{kr} \Delta s_i^{kr}. \end{aligned} \quad (4.6)$$

Otázka teraz znie, či sa nachádzame bližšie k nule oproti výsledku z (4.4), kde sme neuvažovali korekčnú zložku kroku. Korekčná zložka vo väčšine prípadov vylepší celkovú efektívnosť algoritmu.

Kombinovaný centrujúci-korekčný krok  $(\Delta x^{ck}, \Delta y^{ck}, \Delta s^{ck})$  získame riešením systému:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x^{ck} \\ \Delta y^{ck} \\ \Delta s^{ck} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \tau \sigma e - \Delta X^{af} \Delta S^{af} e \end{bmatrix}, \quad (4.7)$$

Koeficienty upravenej matice už máme k dispozícii z riešenia systému (4.3), takže systém (4.7) vyriešime jednoducho pomocou spätnej substitúcie.

Teraz špecifikujeme Mehrotrov algoritmus ako celok, potom čo sme popísali základné elementy Mehrotrových úvah o ňom.

### 4.3.1 Mehrotrov algoritmus

**Vstup:**  $(x^o, y^o, s^o)$ ,  $(x^o, s^o) > 0$   
 pre  $k = 0, 1, 2, \dots$

1. rieš:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^{af} \\ \Delta y^{af} \\ \Delta s^{af} \end{bmatrix} = \begin{bmatrix} -r_b \\ -r_c \\ -XSe \end{bmatrix},$$

2. vyrátaj:

$$\begin{aligned} \alpha_{af}^P &= \max\{\alpha \in [0, 1] \mid x^k + \alpha \Delta x_{af} \geq 0\}, \\ \alpha_{af}^D &= \max\{\alpha \in [0, 1] \mid s^k + \alpha \Delta s_{af} \geq 0\}, \\ \sigma_{af} &= \frac{(x^k + \alpha_{af}^P \Delta x^{af})^T (s^k + \alpha_{af}^D \Delta s^{af})}{n}, \end{aligned}$$

3. prirad:

$$\tau = \left( \frac{\sigma_{af}}{\sigma} \right)^3$$

4. rieš:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^{ck} \\ \Delta y^{ck} \\ \Delta s^{ck} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \tau \sigma e - \Delta X^{af} \Delta S^{af} e \end{bmatrix},$$

5. vyrátaj:

$$\begin{aligned} (\Delta x^k, \Delta y^k, \Delta s^k) &= (\Delta x^{af}, \Delta y^{af}, \Delta s^{af}) + (\Delta x^{ck}, \Delta y^{ck}, \Delta s^{ck}), \\ \alpha_{max}^P &= \max\{\alpha \geq 0 \mid x^k + \alpha \Delta x_k \geq 0\}, \\ \alpha_{max}^D &= \max\{\alpha \geq 0 \mid s^k + \alpha \Delta s_k \geq 0\}, \end{aligned}$$

6. prirad:

$$\begin{aligned} \alpha_k^P &= \min(0, 99 * \alpha_{max}^P, 1), \\ \alpha_k^D &= \min(0, 99 * \alpha_{max}^D, 1), \end{aligned}$$

7. prirad:

$$\begin{aligned} x^{k+1} &= x^k + \alpha_k^P \Delta x^k, \\ (y^{k+1}, s^{k+1}) &= (y^k, s^k) + \alpha_k^D (\Delta y^k, \Delta s^k), \end{aligned}$$

skonči.

# Kapitola 5

## Projekt dizertačnej práce

### 5.1 Súčasný stav problematiky

Karmarkar predstavením svojho algoritmu [11] vyvolal opäť po dlhej dobe záujem o metódy vnútorného bodu. Bolo navrhnuté veľké množstvo algoritmov, z ktorých najjednoduchšie a najefektívnejšie sa ukázali byť primárno–duálne algoritmy sledovania centrálnej trajektórie. Centrálna trajektória sa dostala do popredia záujmu, ako dôležitý element štúdia metód vnútorného bodu.

Kojima, Mizuno a Yoshise [14] v roku 1989, ako jedni z prvých, prezentovali primárno–duálny algoritmus metód sledovania centrálnej trajektórie s krátkym krokom. Algoritmus je najjednoduchším z tejto kategórie a podarilo sa pre počet Newtonovských iterácií v tomto algoritme dokázať polynomiálnu zložitosť  $O(\sqrt{n}L)$ .<sup>1</sup> Analýze metódy sledovania centrálnej trajektórie s krátkym krokom je venovaná Kapitola 3.1 tejto práce.

Paralelne, Monteiro a Adler [22], [23], predstavili algoritmus s krátkym krokom a taktiež dokázali jeho polynomiálnu zložitosť typu  $O(\sqrt{n}L)$ . Neskôr sa zaoberali aj štúdiom prediktor–korektor algoritmu, pre ktorý sa im tiež podarilo dokázať polynomiálnu zložitosť  $O(\sqrt{n}L)$ . Prediktor–korektor<sup>2</sup> algoritmus bol prvýkrát prezentovaný v diele autorov Mizuno, Todd a Ye [21], ako určitá varianta algoritmu s krátkym krokom. Autori však navrhli v jednotlivých iteráciách striedať dva druhy krokov, a to prediktor krok (v zmysle zlepšovania mierky duality) a korektor krok (v zmysle centrality). V tejto práci je jeho analýze venovaná Kapitola 3.2

S ideou rozšíriť metódy úloh lineárneho programovania aj pre neprípustné štartovacie body prišli Lustig, Marsten a Shanno vo svojich prácach [16], [17]. Podobne sa venovali problematike neprípustných bodov Kojima, Megiddo a Mizuno v [13], kde navrhujú dĺžku kroku pre tieto algoritmy s neprípustným štartujúcim bodom

<sup>1</sup>L predstavuje celkovú dĺžku bitového zápisu riešenia úlohy.

<sup>2</sup>V literatúre je často označovaný pod názvom Mizuno–Todd–Ye algoritmus, podľa mien jeho autorov.

tak, aby bola zachovaná globálna konvergencia algoritmu. Navrhovaný algoritmus nájde optimálne riešenie úlohy za predpokladu, že je množina vnútorných bodov neprázdna.

V nadväznosti na túto prácu, Lustig, Marsten a Shanno v [17] publikovali výsledky svojej práce, v ktorej testovali výpočtovú efektívnosť zovšeobecného prediktor–korektor algoritmu, pre ktorý Kojima, Megiddo a Mizuno dokázali globálnu konvergenciu. Numerické testy boli nadmieru pozitívne, čo indikovalo bezpečnosť tohoto algoritmu a to v celku pri malých výpočtových nákladoch. Výsledky numerických testov pre modifikáciu algoritmu pre neprípustné body sa stretli už s menším úspechom.

Mizuno [20] ukázal, že modifikácia algoritmu Kojima, Megiddo a Mizuna rieši dvojicu úloh lineárneho programovania v polynomiálnom čase bez toho, aby sme museli predpokladať existenciu riešenia úlohy lineárneho programovania. Navyše sa mu podarilo pre modifikáciu spomínaného algoritmu ukázať zložitosť  $O(nL)$ .

Analýze Mizuno–Todd–Ye algoritmu (t.j. prediktor–korektor algoritmu) sa venujú Gonzaga a Tapia v [5] a [6], (1997). V nadväznosti na dovtedy zistené poznatky o prediktor–korektor algoritme, pre ktorý bola dokázaná zložitosť  $O(\sqrt{nL})$ , ukázali že algoritmom generovaná postupnosť iterácií (postupnosť pre prediktor krok a postupnosť pre korektor krok) konverguje k analytickému stredmu množiny riešenia úlohy [5] a že konvergencia je kvadratická [6].

Podobne sa v tom istom období venovali štúdiu konvergence a zložitosti prediktor–korektor algoritmu Sheng a Potra [28]. Ukázali, že zložitosť algoritmu závisí od kvality výberu štartovacieho bodu. Pre neprípustné štartovacie body má algoritmus zložitosť  $O(nL)$  a teda pomocou  $O(nL)$  iterácií vie zistiť aj neexistenciu riešenia. Ak je úloha riešiteľná a štartujúci bod je zvolený dostatočne blízko k optimálnemu riešeniu, tak dostávame zložitosť typu  $O(\sqrt{nL})$ .

V nedávnej práci Monteiro a Tsuchiya [24] autori prezentovali nové hranice výpočtovej zložitosti pre Mizuno–Todd–Ye prediktor–korektor algoritmus.

Za veľmi prínosnú prácu je považovaná publikácia Mehrotro [19] ešte z roku 1992, kde sa snažil sklbiť doterajšie poznatky z metód sledovania centrálnej trajektórie a poznatky o neprípustných štartovacích bodoch. Prezentoval vysokoefektívny Mehrotrov prediktor–korektor algoritmus, v ktorom na základe početných numerických testovaní a výpočtov navrhol heuristicky voliť cetrujúci parameter  $\tau$ . Algoritmus sa stal vďaka svojej jednoduchšej štruktúre a dobrým numerickým výsledkom základom pre väčšinu zdrojových kódov pre metódy vnútorného bodu. Konceptia Mehrotrovho algoritmu je predstavená v tejto práci v Kapitole 4. Pre tento algoritmus zatiaľ neexistuje korektný dôkaz zložitosti a je preň otvorených mnoho teoretických otázok. Mnoho autorov sa pokúšalo o implementáciu Mehrotrovho algoritmu, napr. [18], kde sa ukázalo numerickými výpočtami, že algoritmus je efektívny najmä pre úlohy veľkých rozmerov.

Snahu o dokázanie konvergencie a zložitosti Mehrotrovho algoritmu môžeme sledovať aj v práci Y. Zhanga a D. Zhanga [36] z 1995. V Mehrotrovom algoritme sa snažili redukovať vplyv korekčnej zložky tým, že ju vynásobili druhou mocninou dĺžky kroku novej iterácie. Výsledkom týchto snažení boli prediktor–korektor metódy s korekčnou zložkou druhého rádu. Pre navrhnuté dve modifikácie Mehrotrovho algoritmu vedeli určiť odhad polynomiálnej zložitosti.

O rok neskôr publikovali Tapia, Y. Zhang, Saltzman a Weiser prácu [29], v ktorej poukazovali na súvislosti medzi klasickou Newtonovou metódou, ďalej algoritmom, ktorý navrhol Kojima, Mizuno a Yoshise [14] a Mehrotrovým algoritmom, ktorý považovali za istú perturbáciu Newtonovej metódy. Za určitých prísnejších predpokladov, ako sú bežne kladené pri metódach vnútorných bodov, ukázali lokálnu konvergenciu algoritmu.

Ďalšie snahy o rozvoj algoritmov, v rámci primárno–duálnych metód vnútorných bodov, generujúcich smery vyšších rádov možno zaznamenať v práci Jarreho a Wechsa [10], ktorá analyzuje Mehrotrov algoritmus, konkrétne jeho korekčnú zložku a dáva návrh na modifikáciu Mehrotrovho algoritmu v tom zmysle, že povolí niekoľkokrát opakovať korekčnú zložku kroku. Takáto zložka smeru vyššieho rádu v Mehrotrovom algoritme je však v modifikovanom algoritme preformulovaná ako lineárny podprogram, ktorý sa opakuje v každej iterácii. To, že je efektívnosť takto modifikovaného algoritmu prínajmenšom taká dobrá ako pri klasickom Mehrotrovom algoritme, autori potvrdzujú priloženými numerickými výpočtami.

Za najnovšie odborné výsledky z oblasti problematiky môžeme považovať práce C. Cartis [2] a [3], (2004). V práci [3], ako prvá poukazuje na prípady, kedy môže efektívny Mehrotrov algoritmus zlyhať a nekonvergovať. Analyzuje, že väčšina implementácií Mehrotrovho algoritmu, vďaka jeho všeobecne známym dobrým výsledkom správania sa a konvergencie, vôbec neberie do úvahy možnosť zlyhania algoritmu a nemá v sebe žiadne prvky, ktoré by monitorovali konvergenciu a v prípade potreby by sa tak mohli pomôcť vyhnúť situáciám, v ktorých dochádza k zlyhávaniu. Pokúša sa o modifikovanú konštrukciu Mehrotrovho algoritmu, tzv. primárno–duálneho korektor algoritmu, a vysvetľuje správanie sa takto skonštruovaného algoritmu. Poukazuje na prípady, pre aké voľby štartujúcich bodov a pri akých hodnotách centrujúceho parametra dochádza k zlyhaniu. Príčinu vidí v tom, že korekčná zložka má príliš veľký vplyv na jednotlivé iterácie a tým pádom aj značne ovplyvňuje ich smerovanie. Podobne ako v [36] navrhuje znížiť vplyv korektora tak, že ho násobí druhou mocninou dĺžky kroku pre príslušnú iteráciu. Svoje teoretické tvrdenia tu podkladá numerickými experimentami.

Na túto prácu nadväzuje svojou ďalšou prácou [2], kde rozoberá teoretické pozadie navrhnutého primárno–duálneho algoritmu s korekčnou zložkou druhého rádu. Pokúša sa tu ďalej skúmať konvergenciu a zložitost' algoritmu, za predpokladu existencie ostro prípustného štartovacieho bodu. Tak isto skúma najvhodnejšiu voľbu centrujúceho parametra (ktorý by mal byť asymptoticky toho istého rádu

ako parameter mierky duality pre jednotlivé iterácie). Ukazuje tu, že pri zakomponovaní voľby dlhého kroku do algoritmu, postupnosť iterácií konverguje k ostro prípustnému riešeniu úlohy lineárneho programovania, ktoré však nemusí byť analytickým stredom množiny primárno-duálnych riešení. Tvrdenia opäť demonštruje na konkrétnych príkladoch.

Poslednou zatiaľ známou publikáciou, ktorá skúma vlastnosti Mehrotrovho algoritmu je práca Salahiho, Penga a Terlakyho [27]. Autori tu opäť otvárajú otázky dosiaľ nedokázanej polynomiálnej zložitosti Mehrotrovho prediktor-korektor algoritmu. Adaptívna voľba centrujúceho parametra  $\tau$ , ktorá podľa ich zistení môže spôsobiť, že budeme robiť priveľa malých krokov, aby sme sa nachádzali v blízkosti centrálnej trajektórie (čo je nevyhnutné pre dôkaz polynomiálnej zložitosti algoritmu), bola pre nich motiváciou na zakomponovanie "bezpečnostného" prvku do algoritmu. Koordinovali tak minimálnu možnú dĺžku kroku pri zvolenej veľkosti okolia, v ktorom sa majú všetky iterácie nachádzať. Podarilo sa im ukázať, že takto modifikovaný algoritmus potrebuje na konvergenciu v najhoršom prípade  $O(n^2 \log \frac{(x^o)^T s^o}{\epsilon})$  iterácií. Pri ďalšej modifikácii korekčnej zložky algoritmu sa im podarilo dokonca znížiť počet iterácií na  $O(n \log \frac{(x^o)^T s^o}{\epsilon})$ . V takto pozmenenom algoritme navrhli navyše mierne modifikovať Mehrotrom heuristiky volenú veľkosť centrujúceho parametra, čo im implikovalo superlineárnu konvergenciu. Polynomiálnu zložitost' a globálnu konvergenciu pre pôvodný Mehrotrom navrhnutý algoritmus sa nepodarilo dokázať ani týmto autorom, aj keď na druhej strane predostreli jeho verziu s teoretickým matematickým základom.

## 5.2 Prehľad doktorandom dosiahnutých výsledkov

Predložená práca sa pokúša sprostredkovať komplexný prehľad o základných algoritmoch metód sledovania centrálnej trajektórie, medzi ktoré radíme algoritmus s krátkym krokom, prediktor-korektor algoritmus, algoritmus s dlhým krokom a Mehrotrov algoritmus. V práci sú zhrnuté doterajšie teoretické poznatky o týchto algoritmoch. Kapitola 3, ktorá predstavuje nosnú časť práce, venuje svoju pozornosť teoretickej analýze metódy s krátkym krokom, prediktor korektor metódy a metódy s dlhým krokom. Pre každú zo spomínaných metód je uvedený príslušný algoritmus pre nájdenie optimálneho riešenia. V uvádzaných algoritmoch predpokladáme prípustnosť štartujúceho bodu, pričom je zrejmé, že algoritmy môžu byť zovšeobecnené aj pre neprípustné štartovacie body, tzv. studené štarty. Oproti výkladu z [33] sú uvádzané dôkazy polynomiálnej zložitosti, lineárnej konvergenice a tak isto aj ďalších uvádzaných viet a liem podrobnejšie zdôvodnené a vyargumentované. Ako je známe, metóda prediktor–korektor používa dva druhy okolia a to vnútorné ( $\mathcal{N}_2(\theta_{in})$ ) a vonkajšie ( $\mathcal{N}_2(\theta_{out})$ ), kde v práci [33] je pevne zvolené

$\theta_{in} = 0,25$  a  $\theta_{out} = 0,5$ . Predkladaná práca zovšeobecňuje v Kapitole 3.2 celú analýzu prediktor–korektor metódy pre všeobecnú voľbu týchto parametrov a určuje vzťah medzi týmito parametrami tak, aby bola zachovaná funkčnosť a spoľahlivosť algoritmu. V závislosti od všeobecnej voľby parametrov je uvádzaná aj dĺžka kroku  $\alpha$  pre jednotlivé iterácie. Taktiež sa podarilo ukázať, že konkrétne hodnoty parametrov, pôvodne používané vo výklade prediktor–korektor algoritmu, vyhovujú nami odvodeným zovšeobecneným vzťahom.

Mehrotrovmu algoritmu je venovaná samostatná Kapitola 4, ktorá vysvetľuje jeho podstatu a zmysel, ako aj celkovú koncepciu. Pre tento algoritmus však doposiaľ nebola dokázaná polynomiálna zložitosť a ani globálna lineárna konvergencia. Aj napriek chabým teoretickým základom je tento algoritmus vysoko populárny a stal sa východiskom pre mnohé zdrojové kódy programov metód vnútorného bodu.

Predkladaná práca poskytuje teoretický materiál pre ďalšie plánované (a možno i zatiaľ neplánované) aktivity výskumu a implementácie.

### 5.3 Ciele dizertačnej práce

Doterajšie poznatky, ktorých stručný prehľad je vyššie uvedený, nás ďalej motivujú k hlbšiemu štúdiu problematiky a k vytýčeniu si strategických cieľov pre ďalšiu prácu:

- ◇ Nadviazať na analýzu článkov C. Cartis [2], [3] a Salayho, Penga a Terlakyho [27] a hlbšie preskúmať nimi navrhnuté modifikácie Mehrotrovho algoritmu.
- ◇ Pokúsiť sa získať zdrojový kód Mehrotrovho algoritmu, alebo niektorej jeho modifikácie. Veľká časť z nich je voľne prístupná, napr. LOQO, založený na Mehrotrovom algoritme, písaný v jazyku C, by mal byť voľne dostupný na účely výskumu na <http://www.orfe.princeton.edu/~loqo>.
- ◇ Numerickými experimentami otestovať získaný softvér pre rôzne voľby hodnôt vstupných parametrov.
- ◇ Preskúmať najlepšie fungovanie algoritmu pre rozličné triedy úloh.
- ◇ Aplikovať Mehrotrov algoritmus, prípadne niektorú z jeho modifikácií, na riešenie optimalizačných úloh z finančnej oblasti, najmä na úlohy veľkých rozmerov.



# Literatúra

- [1] Boyd, S. – Vandenberghe, L.: *Convex Optimization*, Cambridge University Press, [http://www.stanford.edu/~boyd/bv\\_cvxbook.pdf](http://www.stanford.edu/~boyd/bv_cvxbook.pdf), (2004)
- [2] Cartis, C.: *On the Convergence of a Primal–Dual Second–Order Corrector Interior Point Algorithm for Linear Programming*, <http://www.optimization-online.org>, (2005)
- [3] Cartis, C.: *Some disadvantages of a Mehrotra–Type Primal–Dual Corrector Interior Point Algorithm for Linear Programming*, <http://www.optimization-online.org>, (2005)
- [4] Gill, P. – Murray, W. – Saunders, M. – Tomlin, J. – Wright, M.: *On projected Newton barrier methods for linear programming and an equivalence to Karmarkar’s projective method.*, *Mathematical Programming*, Vol. 36, 183–209, (1986)
- [5] Gonzaga, C.C. – Tapia, R.A.: *On the Convergence of the Mizuno–Todd–Ye Algorithm to the Analytic Center of the Solution Set*, *SIAM Journal on Optimization*, Vol. 7, No. 1, 47–65, (1997)
- [6] Gonzaga, C.C. – Tapia, R.A.: *On the Quadratic Convergence of the Simplified Mizuno–Todd–Ye Algorithm for the Linear Programming*, *SIAM Journal on Optimization*, Vol. 7, No. 1, 66–85, (1997)
- [7] Halická, M.: *Dvadsať rokov moderných metód vnútorného bodu*, *Pokroky matematiky, fyziky a astronómie*, ročník 49, č. 3, 234–244, (2004)
- [8] Halická, M.: *Metódy vnútorného bodu*, seminár (2004)
- [9] Hamala, M.: *Nelineárne programovanie*, seminár (2003)
- [10] Jarre, F. – Wechs, M.: *Extending Mehrotra’s Corrector for Linear Programs*, *Advanced Modeling and Optimization*, Vol. 1, No. 2, (1999)
- [11] Karmarkar, N.: *A new polynomial–time algorithm for linear programming*, *Combinatorica*, Vol. 4, 373–395, (1984)

- [12] Klee, V. – Minty, G.: *How good is the simplex algorithm?* In: O. Shisha, ed., "Inequalities–III", Academic Press, New York, (1972)
- [13] Kojima, M. – Megiddo, N. – Mizuno, S.: *A primal–dual infeasible–interior–point algorithm for linear programming*, Mathematical Programming, Vol. 61, 263–280, (1993)
- [14] Kojima, M. – Mizuno, S. – Yoshise, A.: *A polynomial–time algorithm for a class of linear complementarity problems*, Mathematical Programming, Vol. 44, 1–26, (1989)
- [15] Kojima, M. – Shida, M. – Shindoh, S.: *Local convergence of predictor–corrector infeasible–interior–point algorithms for SDPs and SDLCPs*, Mathematical Programming, Vol. 80, 129–160, (1998)
- [16] Lustig, I.J. – Marsten, R.E. – Shanno, D.F.: *Computational experience with a primal–dual interior–point method for linear programming*, Linear Algebra and Its applications, Vol. 152, 191–222, (1991)
- [17] Lustig, I.J. – Marsten, R.E. – Shanno, D.F.: *Computational experiences with a globally convergent primal–dual predictor–corrector algorithm for linear programming*, Mathematical Programming, Vol. 66, 123–135, (1994)
- [18] Lustig, I. – Marsten, R. – Shanno, D.: *On Implementing Mehrotra’s Predictor–Corrector Interior–Point Method for Linear Programming*, SIAM Journal on Optimization, Vol. 2, No. 3, 435–449 (1992)
- [19] Mehrotra, S.: *On the implementation of a primal–dual interior point method*, SIAM Journal on Optimization, Vol. 2, No. 4, 575–601, (1992)
- [20] Mizuno, S.: *Polynomiality of infeasible–interior–point algorithms for linear programming*, Mathematical Programming, Vol. 67, 109–119, (1994)
- [21] Mizuno, S. – Todd, M.J. – Ye, Y.: *On adaptive–step primal–dual interior–point algorithms for linear programming*, Mathematics of Operations Research, Vol. 18, 964–981, (1993)
- [22] Monteiro, R.D.C. – Adler, I.: *Interior path following primal–dual algorithms. part I: Linear programming*, Mathematical Programming, Vol. 44, 27–41, (1989)
- [23] Monteiro, R.D.C. – Adler, I.: *Interior path following primal–dual algorithms. part II: Convex quadratic programming*, Mathematical Programming, Vol. 44, 43–66, (1989)

- [24] Monteiro, R.D.C. – Tsuchiya, T.: *A new iteration–complexity bound for the MTY predictor–corrector algorithm*, SIAM Journal on Optimization, Vol. 15, No. 2, 319–347, (2004)
- [25] Plesník, J. – Dupačová, J. – Vlach, M.: *Lineárne programovanie*, Alfa, Bratislava (1990)
- [26] Potra, F.A. – Wright, S.J.: *Interior–Point Methods*, Journal of Computation and Applied Mathematics, Vol. 124, 281–302, (2000)
- [27] Salahi, M. – Peng, J. – Terlaky, T.: *On Polynomial Complexity of Mehrotra–Type Predictor–Corrector Algorithms*, <http://www.optimization-online.org>, (2005)
- [28] Sheng, R. – Potra, F.A.: *A Quadratically Convergent Infeasible–Interior–Point Algorithm for LCP with Polynomial Complexity*, SIAM Journal on Optimization, Vol. 7, No. 2, 304–317, (1997)
- [29] Tapia, R. – Zhang, Y. – Saltzman, M. – Weiser, A.: *The Mehrotra Predictor–Corrector Interior–Point Method As a Perturbed Composite Newton Method*, SIAM Journal on Optimization, Vol. 6, No. 1, 47–56 (1996)
- [30] Tapia, R.A. – Zhang, Y. – Ye, Y.: *On the convergence of the iteration sequence in primal–dual interior point methods*, Mathematical Programming, Vol. 68, 141–154, (1995)
- [31] Vanderbei, R.J.: *Linear Programming: Foundations and Extensions*, Princeton, <http://www.princeton.edu/~rvdb/LPbook/>, (2001)
- [32] Wright, M.H.: *The Interior–Point Revolution in Optimization: History, Recent Developments, and Lasting Consequences*, Bulletin of the American Mathematical Society, Vol. 42, No. 1, 39–56, (2004)
- [33] Wright, S.J.: *Primal–Dual Interior–Point Methods*, Society for Industrial and Applied Mathematics, Philadelphia (1997)
- [34] Ye, Y. – Güler, O. – Tapia, R.A. – Zhang, Y.: *A quadratically convergent  $O(\sqrt{n}L)$ –iteration algorithm for linear programming*, Mathematical Programming, Vol. 59, 151–162, (1993)
- [35] Zhang, F.: *Matrix Theory, Basic Results and Techniques*, Springer–Verlag New York (1999)
- [36] Zhang, Y. – Zhang, D.: *On polynomiality of the Mehrotra–type predictor–corrector interior–point algorithms*, Mathematical Programming, Vol. 68, No. 3, 303–318, (1995)