

UNIVERZITA PALACKÉHO V OLOMOUCI  
PŘÍRODOVĚDECKÁ FAKULTA  
KATEDRA MATEMATICKÉ ANALÝZY A APLIKACÍ MATEMATIKY

## DIPLOMOVÁ PRÁCE

Řešení úloh nelineárního programování pomocí  
primárně-duální metody nelineárního rescalingu



Vedoucí diplomové práce:  
**RNDr. Horymír Netuka, Ph.D.**  
Rok odevzdání: 2012

Vypracoval:  
**Bc. Richard Andrášik**  
MAP, V. ročník

### **Prohlášení**

Prohlašuji, že jsem vytvořil tuto diplomovou práci samostatně za vedení a pomoci RNDr. Horymíra Netuky, Ph.D. a že jsem v seznamu použité literatury uvedl všechny zdroje použité při zpracování práce.

V Olomouci dne 23. března 2012

## **Poděkování**

Rád bych na tomto místě poděkoval RNDr. Horymíru Netukovi, Ph.D. jako svému vedoucímu diplomové práce za obětavou spolupráci i za čas, který mi věnoval při konzultacích, a svým rodičům za podporu po celou dobu mého studia. Také děkuji RNDr. Pavlu Ženčákovi, Ph.D. za poskytnutí materiálů k aplikačním příkladům.

# Obsah

<b>Použité symboly</b>	<b>4</b>
<b>Úvodní slovo</b>	<b>5</b>
<b>1 Přípravná kapitola</b>	<b>8</b>
1.1 Základy teorie nelineárního programování . . . . .	9
1.1.1 Lagrangeova funkce a lagrangeovská dualita . . . . .	9
1.1.2 Podmínky optimality . . . . .	11
1.2 Newtonova metoda . . . . .	13
<b>2 Základní myšlenky nelineárního rescalingu</b>	<b>14</b>
2.1 Zadání problému . . . . .	14
2.2 Funkce nelineárního rescalingu . . . . .	16
2.3 Ekvivalentní problém . . . . .	19
2.4 Metoda nelineárního rescalingu . . . . .	22
2.5 Primárně-duální metoda nelineárního rescalingu . . . . .	23
2.6 Konvergence metod NR a PDNR . . . . .	30
<b>3 Vylepšení metod založených na NR</b>	<b>35</b>
3.1 Dynamická změna parametru rescalingu . . . . .	35
3.2 Volba délky kroku . . . . .	37
3.3 Primárně-duální metoda sledování cesty . . . . .	43
3.3.1 Metoda sledování cesty . . . . .	43
3.3.2 Newtonova metoda pro primárně-duální systém . . . . .	45
3.3.3 Kombinace přístupu NR a PDPF metody . . . . .	46
3.4 Regularizovaná Newtonova metoda . . . . .	46
3.4.1 Použití RNM v teorii NR . . . . .	51
<b>4 Příklady</b>	<b>54</b>
4.1 O parametrech PDNRD metody . . . . .	54
4.2 Problém se strunou . . . . .	57
4.3 Problém s ocelovým kvádrem . . . . .	61
<b>Závěr</b>	<b>66</b>
<b>Literatura</b>	<b>67</b>

## Použité symboly

$\mathbb{N}$	obor přirozených čísel
$\mathbb{N}_0$	$\mathbb{N}_0 = \mathbb{N} \cup \{0\}$
$\mathbb{R}$	obor reálných čísel
$\mathbb{R}_+$ ( $\mathbb{R}_{++}$ )	obor nezáporných (resp. kladných) reálných čísel
$\mathbb{R}^n$	vektorový prostor dimenze $n$ nad $\mathbb{R}$
$\mathbb{R}^{n \times m}$	prostor matic typu $(n, m)$ nad $\mathbb{R}$
$(a, b)$	otevřený interval
$\langle a, b \rangle$	uzavřený interval
$A = (a_{ij})_{i,j=1}^n$	čtvercová matice řádu $n$
$A^T$	transponovaná matice
$\mathbf{I}_n$	jednotková matice řádu $n$
$\mathbf{O}_n$	nulová matice řádu $n$
$x = (x_1, \dots, x_n)^T$	uspořádaná $n$ -tice reálných čísel – vektor
$\text{diag}(x)$	diagonální matice s prvky vektoru $x$ na diagonále
$\mathcal{D}(f)$	definiční obor funkce $f$
$\mathcal{C}^n(\mathcal{M})$	množina všech funkcí, které mají spojitě všechny derivace na množině $\mathcal{M}$ až do řádu $n$ , kde $n \in \mathbb{N}_0$
$\nabla f(x)$	gradient funkce $f$ v bodě $x$
$\nabla^2 f(x)$	Hessova matice funkce $f$ v bodě $x$
$\nabla F(x)$	Jacobiho matice zobrazení $F$ v bodě $x$
□	konec důkazu
■	konec řešení příkladu

## Úvodní slovo

Matematika za několik tisíciletí své existence urazila dlouhou a velice náročnou cestu, při níž se její tvář mnohokrát změnila, avšak její podstata zůstala stejná. Je to nejspíše kvůli různým pohledům ve vnímání přírody, u níž se matematikové inspirovali, a stejnému cíli – porozumět světu okolo nás.

*„Velká kniha přírody je psána matematickou řečí.“*

*Galileo Galilei*

Když čelíme nějakému problému, máme zpravidla širokou škálu možností, jak jej vyřešit. Každý z nás si jistě někdy položil otázku: „Nemohl bych to udělat nějak lépe?“ nebo „Jaký by byl nejlepší způsob?“ Je zřejmé, že vyzkoušet všechny možnosti není nejlepší přístup. Snad nikoho nepřekvapí, že se matematika snaží odpovědět i na tyto otázky. Přesněji řečeno odpovědi hledá teorie optimalizace.

*„Ve světě se neděje nic, aniž by v tom nehrál roli zákon optimalizace.“*

*Leonhard Euler*

Optimalizace je matematická disciplína zabývající se studiem problémů a hledající nejlepší možné řešení. Nejdříve se převede zadaný problém do řeči matematiky. Udělá se to tak, že se definuje vhodná funkce, která popisuje daný jev či problém. Takovéto funkci se říká účelová. Dále se určí podmínky, zpravidla ve tvaru rovnic a nerovnic, které omezují množinu řešení. Následně je potřeba zjistit, ve kterém bodě, jenž vyhovuje podmínkám, má účelová funkce své minimum respektive maximum. Hledají se tedy extrémní účelové funkce.

Na příklady použití matematické optimalizace narazíme snad ve všech vědních oborech fyzikou (vývoj automobilů a letadel, stavba mostů, transport elektrické energie) a chemií (optimalizace složení vody) počínaje přes ekonomickou sféru (složení portfolia, logistické problémy) až k medicíně (plánování radiační terapie pro léčbu rakoviny – viz [1]).

Teorie optimalizace se vyvíjela souběžně s výskytem aplikačních problémů a potřebou tyto problémy řešit. Díky tomu některé typy úloh získaly určité speciální postavení mezi ostatními. Jedná se například o úlohu lineárního programování (LP).

Typickou úlohou LP je najít nejlevnější způsob, jak dodat zboží z několika skladů s danými dodávkami do souboru prodejen s danými požadavky. V roce 1947 George Dantzig uvedl simplexovou metodu pro řešení LP (viz [5]). V současné době má ovšem simplexová metoda silného konkurenta – metody vnitřních bodů (viz [16]).

Na poli nelineárního programování (NLP) se rovněž uplatňují metody vnitřních bodů, avšak zde není situace tak jednoduchá jako u LP a metody vnitřních bodů se zde někdy setkávají s numerickými obtížemi. Tato skutečnost motivovala Romana Polyaka a Igora Grivu k vytvoření metody nelineárního rescalingu.

Cílem této práce je seznámit čtenáře s teorií nelineárního rescalingu (NR) jako nástrojem pro řešení úloh podmíněné optimalizace. První kapitola je věnována základním pojmům nutným k porozumění textu.

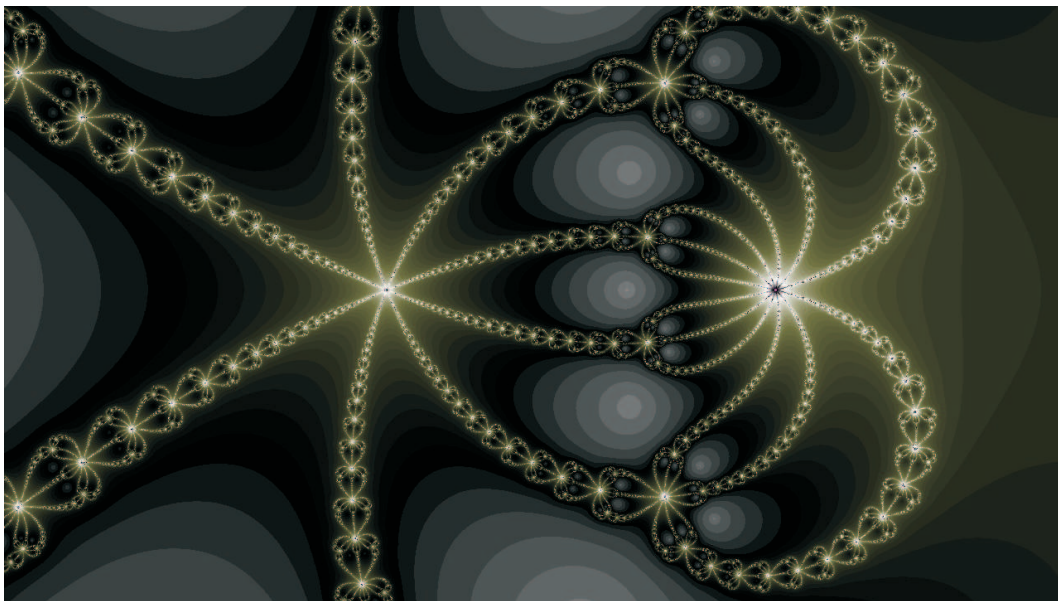
Ve druhé kapitole jsou prezentovány základní myšlenky teorie NR, tj. převod původní úlohy na úlohu ekvivalentní pomocí funkce nelineárního rescalingu, následná analýza této úlohy a popis klasické metody nelineárního rescalingu a její primárně-duální varianty. Rovněž je zde zaměřena pozornost na analýzu konvergence.

Třetí kapitola se zabývá vylepšením metod založených na teorii NR. Je zde diskutována otázka zvýšení rychlosti konvergence metody, dále pak dosažení globální konvergence a použití metody na nehladké funkce.

Ve čtvrté kapitole je na příkladech testována primárně-duální metoda nelineárního rescalingu s dynamickou změnou parametru (PDNRD). Na základě tohoto testování jsou učiněny závěry o chování této metody a také o vhodném nastavení parametrů metody.

Kromě uceleného pohledu na teorii NR je přínosem této práce programová realizace algoritmů ve výpočetním systému MATLAB. Tyto M-soubory, jež byly

vytvořeny za účelem testování metod a řešení ilustračních příkladů, jsou přiloženy na CD. V kapitole 3.4.1 je popsáno použití regularizované Newtonovy metody v teorii NR. Díky tomuto lze metodiku NR zobecnit na nehladké funkce, což je druhým přínosem této práce. Třetím přínosem je vysvětlení významu jednotlivých parametrů PDNRD metody, testování této metody a následný popis jejího chování (viz kapitola 4).



Obrázek 0: Rozmanitost matematiky.



# 1 Přípravná kapitola

Tato kapitola se zabývá vysvětlením některých důležitých pojmů, jež jsou nezbytné k porozumění následujícímu textu. Zejména pojednává o základech teorie podmíněné optimalizace. Jsou zde uvedeny nutné a postačující podmínky optimality, rovněž také kvalifikační podmínky a další potřebné pojmy.

Nechť je dána funkce  $f : \mathcal{M} \rightarrow \mathbb{R}$ , kde  $\mathcal{M} \subseteq \mathbb{R}^n$ . Hledáme extrémy funkce  $f$  na množině  $\mathcal{M}$ . Poněvadž je maximum funkce  $f$  stejné jako mínus minimum funkce  $-f$ , postačí, když dále budeme mluvit jen o minimu funkce, pro maximum funkce platí analogické úvahy a tvrzení.

Víme tedy již, že hledáme minimum funkce na dané množině  $\mathcal{M}$ . Ideální je najít globální minimum této funkce.

**Definice 1.1.** Nechť je dána funkce  $f : \mathcal{M} \rightarrow \mathbb{R}$ ,  $\mathcal{M} \subseteq \mathbb{R}^n$ ,  $x^* \in \mathcal{M}$  nazveme bodem **globálního minima**, jestliže

$$f(x^*) \leq f(x), \quad \forall x \in \mathcal{M}.$$

Pokud navíc platí  $f(x^*) < f(x)$  pro všechna  $x \in \mathcal{M}$ ,  $x \neq x^*$ , bod  $x^*$  nazýváme bodem **ostrého** globálního minima funkce  $f$ .

Obvykle je velmi těžké najít globální minimum funkce, poněvadž zpravidla nevíme, jak funkce vypadá na celé množině  $\mathcal{M}$ . Známe pouze její průběh v určitých lokálních oblastech, a tedy pomocí většiny numerických metod jsme schopni najít pouze lokální minimum dané funkce.

**Definice 1.2.** Nechť je dána funkce  $f : \mathcal{M} \rightarrow \mathbb{R}$ ,  $\mathcal{M} \subseteq \mathbb{R}^n$ ,  $x^* \in \mathcal{M}$  nazveme bodem **lokálního minima**, jestliže existuje okolí  $\mathcal{U}$  bodu  $x^*$  takové, že

$$f(x^*) \leq f(x), \quad \forall x \in \mathcal{U} \cap \mathcal{M}.$$

Takovéto lokální minimum nazýváme **neostré**. Pokud navíc platí  $f(x^*) < f(x)$  pro všechna  $x \in \mathcal{U} \cap \mathcal{M}$ ,  $x \neq x^*$ , bod  $x^*$  nazýváme bodem **ostrého** lokálního minima funkce  $f$ .

## 1.1 Základy teorie nelineárního programování

Úlohu podmíněné optimalizace můžeme zapsat ve tvaru:

$$\begin{cases} \text{minimalizovat} & f(x), \quad x \in X \subseteq \mathbb{R}^n, \\ \text{za podmíněk} & c_i(x) \geq 0, \quad i = 1, \dots, r, \\ & d_j(x) = 0, \quad j = 1, \dots, s. \end{cases} \quad (1.1)$$

Minimalizovanou funkci  $f(x)$  nazýváme **účelovou funkcí** nebo **cenou**. Prvkům  $x \in X$ , pro které jsou splněny podmínky uvedené v (1.1), říkáme **přípustné body**. V dalším textu budeme předpokládat, že všechny funkce uvedené v (1.1) jsou alespoň třídy  $\mathcal{C}^2$  na množině  $X$ .

Pro libovolný přípustný bod  $\hat{x}$  definujme indexovou množinu

$$I(\hat{x}) = \{i \in \{1, \dots, r\}; c_i(\hat{x}) = 0\}. \quad (1.2)$$

Podmínky  $c_i(\hat{x})$ ,  $i \in I(\hat{x})$ , se nazývají **aktivní** v bodě  $\hat{x}$ , ostatní podmínky jsou **neaktivní**.

Speciálními případy (1.1) jsou úlohy LP, kde minimalizovaná funkce i všechny podmínky jsou lineární. Dále úlohy kvadratického programování (QP), ve kterých je situace stejná jako v LP, jen minimalizovaná funkce je kvadratická, a úlohy konvexního programování (KP), kde je účelová funkce konvexní, podmínky ve tvaru rovnosti jsou lineární a podmínky ve tvaru nerovnosti jsou funkce konkávní.

### 1.1.1 Lagrangeova funkce a lagrangeovská dualita

Hlavním nástrojem při analýze optimalizačního problému je k němu příslušející Lagrangeova funkce.

**Definice 1.3.** K úloze (1.1) přiřadíme **Lagrangeovu funkci**, která je definovaná vztahem

$$\mathcal{L}(x; \lambda, \mu) = f(x) - \sum_{i=1}^r \lambda_i c_i(x) + \sum_{j=1}^s \mu_j d_j(x). \quad (1.3)$$

Lagrangeova funkce sice nemá extrémy, avšak může mít sedlové body.

**Definice 1.4.** Bod  $(x^*, \lambda^*, \mu^*) \in X \times \mathbb{R}_+^r \times \mathbb{R}^s$  nazveme **sedlovým bodem** Lagrangeovy funkce příslušné k (1.1), jestliže jsou splněny nerovnosti

$$\begin{aligned} \mathcal{L}(x^*; \lambda, \mu) \leq \mathcal{L}(x^*; \lambda^*, \mu^*) \leq \mathcal{L}(x; \lambda^*, \mu^*), \\ \forall x \in X, \lambda \in \mathbb{R}_+^r, \mu \in \mathbb{R}^s. \end{aligned} \quad (1.4)$$

**Věta 1.1.** (*optimalita sedlového bodu*)

Trojice  $(x^*, \lambda^*, \mu^*) \in X \times \mathbb{R}_+^r \times \mathbb{R}^s$  je sedlovým bodem funkce  $\mathcal{L}(x; \lambda, \mu)$  právě tehdy, když platí

$$\left\{ \begin{array}{ll} \mathcal{L}(x^*; \lambda^*, \mu^*) = \min_{x \in X} \mathcal{L}(x; \lambda^*, \mu^*), & (\text{podmínka minima}) \\ c_i(x^*) \geq 0, \quad \forall i = 1, \dots, r. & (\text{podmínky přípustnosti}) \\ d_j(x^*) = 0, \quad \forall j = 1, \dots, s. & (\text{podmínky komplementarity}) \\ \lambda_i^* c_i(x^*) = 0, \quad \forall i = 1, \dots, r. & (\text{podmínky komplementarity}) \end{array} \right. \quad (1.5)$$

**Důkaz:** viz [3] str. 272.

Mít k dispozici sedlový bod Lagrangeovy funkce je velice užitečné, neboť platí následující věta.

**Věta 1.2.** (*postačitelnost sedlového bodu*)

Je-li  $(x^*, \lambda^*, \mu^*) \in X \times \mathbb{R}_+^r \times \mathbb{R}^s$  sedlovým bodem funkce  $\mathcal{L}(x; \lambda, \mu)$ , pak  $x^*$  je řešením úlohy (1.1).

**Důkaz:** Tvrzení plyne přímo z definice sedlového bodu a věty 1.1.

Lagrangeovu funkci si lze představit jako „můstek“ spojující podmíněnou a nepodmíněnou optimalizaci. Ovšem je nutné upozornit na skutečnost, že existence minima účelové funkce obecně neimplikuje existenci sedlového bodu Lagrangeovy funkce.

**Definice 1.5.** Bod  $\hat{x} \in X$  se nazývá **Slaterův bod** úlohy (1.1), jestliže platí

$$\begin{aligned} c_i(\hat{x}) > 0, \quad \forall i = 1, \dots, r, \\ d_j(\hat{x}) = 0, \quad \forall j = 1, \dots, s. \end{aligned} \quad (1.6)$$

Pokud existuje aspoň jeden Slaterův bod úlohy (1.1), pak říkáme, že je splněna **Slaterova kvalifikační podmínka**.

Podrobněji se lze s kvalifikačními podmínkami seznámit například ve [3].

**Věta 1.3.** *Nechť je dána úloha konvexního programování splňující Slaterovu kvalifikační podmínku. Potom za předpokladu, že existuje řešení  $x^*$  této úlohy, existují také vektory multiplikátorů  $\lambda^* \in \mathbb{R}_+^r$ ,  $\mu^* \in \mathbb{R}^s$  takové, že bod  $(x^*, \lambda^*, \mu^*)$  je sedlovým bodem odpovídající Lagrangeovy funkce  $\mathcal{L}(x; \lambda, \mu)$ .*

**Důkaz:** viz [6] str. 532.

**Definice 1.6.** Úlohu (1.1) označujeme jako **primární úlohu**. Definujeme **duální funkci** předpisem

$$\theta(\lambda, \mu) = \inf_{x \in X} \mathcal{L}(x; \lambda, \mu). \quad (1.7)$$

Úlohu maximalizovat funkci  $\theta(\lambda, \mu)$  za podmínek  $\lambda_i \geq 0$ ,  $\forall i = 1, \dots, r$ , nazýváme **duální úlohou** příslušnou k úloze primární.

**Definice 1.7.** Předpokládejme, že bod  $x^* \in X$  je řešení primární úlohy a dvojice  $(\lambda^*, \mu^*) \in \mathbb{R}_+^r \times \mathbb{R}^s$  je řešení úlohy duální. Rozdíl

$$f(x^*) - \theta(\lambda^*, \mu^*) \quad (1.8)$$

nazveme **dualitní mezerou**.

**Poznámka 1.1.** Lze ukázat (viz [4], str. 234), že v případě úlohy KP platnost Slaterovy kvalifikační podmínky implikuje rovnost  $f(x^*) = \theta(\lambda^*, \mu^*)$ , což nám říká, že dualitní mezera je v tomto případě nulová.

Další informace o lagrangeovské dualitě lze získat například z [4].

### 1.1.2 Podmínky optimality

**Definice 1.8.** Nechť je dána úloha (1.1) a přípustný bod  $\hat{x}$ . Řekneme, že v bodě  $\hat{x}$  jsou splněny **Karush-Kuhn-Tuckerovy podmínky** (KKT), pokud existuje dvojice vektorů  $(\lambda, \mu) \in \mathbb{R}^r \times \mathbb{R}^s$  takových, že platí

$$\begin{cases} \nabla_x \mathcal{L}(\hat{x}; \lambda, \mu) = 0, & \text{(podmínka minima)} \\ \lambda \geq 0, & \text{(podmínka nezápornosti)} \\ \lambda_i c_i(\hat{x}) = 0, \forall i = 1, \dots, r. & \text{(podmínky komplementarity)} \end{cases} \quad (1.9)$$

**Věta 1.4.** (*KKT nutné podmínky prvního řádu*)

Předpokládejme, že přípustný bod  $x^* \in X$  je bodem lokálního minima úlohy (1.1). Dále necht' je splněna Slaterova kvalifikační podmínka. Potom jsou v bodě  $x^*$  splněny KKT podmínky.

**Důkaz:** viz [8] str. 329.

**Věta 1.5.** (*KKT postačující podmínky prvního řádu*)

Necht' je dána úloha KP. Jsou-li v přípustném bodě  $x^* \in X$  splněny KKT podmínky, je bod  $x^*$  globálním minimem úlohy KP.

**Důkaz:** viz [8] str. 346.

**Věta 1.6.** (*nutná podmínka druhého řádu*)

Předpokládejme, že přípustný bod  $x^* \in X$  je bodem lokálního minima úlohy (1.1) a necht' je splněna Slaterova kvalifikační podmínka. Potom jsou v bodě  $x^*$  splněny KKT podmínky a navíc  $\forall p \in \mathbb{R}^n$  takové, že  $\nabla c_i(x^*)^T p \geq 0$ ,  $\forall i \in I(x^*)$ ,  $\nabla d_j(x^*)^T p = 0$ ,  $\forall j = 1, \dots, s$ , platí vztah

$$p^T \nabla_{xx}^2 \mathcal{L}(x^*; \lambda, \mu) p \geq 0. \quad (1.10)$$

**Důkaz:** viz [8] str. 332.

**Věta 1.7.** (*postačující podmínka druhého řádu*)

Předpokládejme, že přípustný bod  $x^* \in X$  společně s dvojicí  $(\lambda, \mu) \in \mathbb{R}^r \times \mathbb{R}^s$  splňuje KKT podmínky. Označme

$$\begin{aligned} I^+(x^*) &= \{i \in I(x^*); \lambda_i > 0\}, \\ I^0(x^*) &= \{i \in I(x^*); \lambda_i = 0\}. \end{aligned}$$

Dále předpokládejme, že každý vektor  $p \neq 0$ , který splňuje

$$\begin{aligned} \nabla c_i(x^*)^T p &= 0, \forall i \in I^+(x^*), \\ \nabla c_i(x^*)^T p &\geq 0, \forall i \in I^0(x^*), \\ \nabla d_j(x^*)^T p &= 0, \forall j = 1, \dots, s, \end{aligned}$$

rovněž splňuje nerovnost

$$p^T \nabla_{xx}^2 \mathcal{L}(x^*; \lambda, \mu) p > 0. \quad (1.11)$$

Potom je bod  $x^*$  bodem ostrého lokálního minima úlohy (1.1).

**Důkaz:** viz [8] str. 333.

## 1.2 Newtonova metoda

Newtonova metoda je infinitní metodou druhého řádu sloužící k řešení soustav nelineárních rovnic. Chceme-li tedy řešit soustavu

$$F(x) = 0, \quad \text{kde } F : \mathbb{R}^n \rightarrow \mathbb{R}^n, \quad (1.12)$$

konstruujeme posloupnost  $\{x_k\}_{k=0}^{+\infty} \subset \mathbb{R}^n$  odhadů řešení soustavy. Pokud máme k dispozici odhad  $x_k$ ,  $k \in \mathbb{N}$ , pak jeden krok Newtonovy metody, za předpokladu existence Jacobiho matice zobrazení  $F$  v bodě  $x_k$ , obnáší výpočet odhadu  $x_{k+1}$  ze vztahů

$$\begin{aligned} \nabla F(x_k) \Delta x &= -F(x_k), \\ x_{k+1} &= x_k + \Delta x, \end{aligned} \quad (1.13)$$

kde vektor  $\Delta x \in \mathbb{R}^n$  se nazývá **Newtonův krok**.

Je důležité poznamenat, že Newtonova metoda hraje v současném vývoji optimalizačních metod pro úlohu NLP klíčovou roli, zejména je využívána v metodách vnitřních bodů.

Newtonovu metodu lze rovněž chápat jako metodu sloužící k řešení úloh nepodmíněné optimalizace. Při hledání minima účelové funkce  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  aplikujeme Newtonovu metodu na soustavu rovnic  $\nabla f(x) = 0$ , tedy budeme konstruovat posloupnost  $\{x_k\}_{k=0}^{+\infty} \subset \mathbb{R}^n$  odhadů řešení optimalizační úlohy podle vztahů

$$\begin{aligned} \nabla^2 f(x_k) \Delta x &= -\nabla f(x_k), \\ x_{k+1} &= x_k + \Delta x, \end{aligned} \quad (1.14)$$

při známém odhadu  $x_k$ ,  $k \in \mathbb{N}$ .

Vzhledem k tomu, že je Newtonova metoda pouze lokálně konvergentní metodou, není situace tak snadná, jak by se mohlo zdát. Podrobně je problematika Newtonovy optimalizační metody popsána ve [2] a [4].

## 2 Základní myšlenky nelineárního rescalingu

Nejdůležitějšími aspekty numerické metody pro řešení úlohy NLP jsou numerická stabilita (metoda nesmí selhat), rychlost konvergence a přesnost. Metoda, kterou navrhli Roman Polyak a Igor Griva, tyto důležité vlastnosti má.

Jedná se o metodu založenou na principu nelineárního rescalingu (NR). Hlavní myšlenka této metody spočívá v tom, že se nahradí původní problém ekvivalentní úlohou, jež má stejné řešení. Ekvivalentní úloha se získá nelineárním rescalingem podmínek, přičemž k nelineárnímu rescalingu se použije funkce  $\psi : \mathbb{R} \rightarrow \mathbb{R}$  určitých vlastností.

V této kapitole je podrobně popsán teoretický přístup k řešení úlohy konvexního programování pomocí primárně-duální varianty metody nelineárního rescalingu.

### 2.1 Zadání problému

V následujícím textu budeme řešit úlohu konvexního programování. Lze ji zadat ve tvaru

$$\begin{cases} \text{minimalizovat} & f(x), \quad x \in \mathbb{R}^n, \\ \text{za podmíněk} & c_i(x) \geq 0, \quad i = 1, \dots, r, \\ & d_j(x) = 0, \quad j = 1, \dots, s. \end{cases} \quad (2.1)$$

Funkce  $c_i(x)$  jsou konkávní,  $\forall i = 1, \dots, r$ . Podmínky ve tvaru rovnosti jsou lineární. Můžeme tedy psát  $d_j(x) = a_j^T x - b_j$ , přičemž  $a_j \in \mathbb{R}^n$ ,  $b_j \in \mathbb{R}$ ,  $\forall j = 1, \dots, s$ .

Podle [4] str. 132 lze lineární podmínky ve tvaru rovnosti eliminovat, díky čemuž postačí, když dále budeme uvažovat úlohu

$$\begin{cases} \text{minimalizovat} & f(x), \quad x \in \mathbb{R}^n, \\ \text{za podmíněk} & c_i(x) \geq 0, \quad i = 1, \dots, r. \end{cases} \quad (2.2)$$

**Poznámka 2.1.** Pro zjednodušení zápisu můžeme někdy z funkcí  $c_i$ ,  $i = 1, \dots, r$ , sestavit zobrazení  $c : \mathbb{R}^n \rightarrow \mathbb{R}^r$ , přičemž

$$c(x) = (c_1(x), c_2(x), \dots, c_r(x))^T, \quad \forall x \in \mathbb{R}^n.$$

Symbol  $\nabla c(x)$  pak značí Jacobiho matici zobrazení  $c$  v bodě  $x$ .

**Definice 2.1.** Definujme **přípustnou množinu** úlohy (2.2) předpisem

$$S = \{x \in \mathbb{R}^n; c_i(x) \geq 0, \forall i = 1, \dots, r\}. \quad (2.3)$$

**Poznámka 2.2.** Díky nerovnosti

$$g(\alpha x + (1 - \alpha)y) \geq \alpha g(x) + (1 - \alpha)g(y), \quad \forall x, y \in \mathbb{R}^n, \forall \alpha \in (0; 1),$$

jež platí pro všechny konkávní funkce  $g$ , je zřejmé, že množina  $S$  je konvexní.

Budeme předpokládat, že:

- (A) Funkce  $f, c_i, \forall i = 1, \dots, r$ , jsou alespoň dvakrát spojitě diferencovatelné na množině  $\mathbb{R}^n$ .
- (B) Množina  $X^*$  všech řešení úlohy (2.2) je neprázdná a omezená.
- (C) Je splněna Slaterova kvalifikační podmínka (viz definice 1.5).

K úloze (2.2) sestavíme Lagrangeovu funkci

$$L(x; \lambda) = f(x) - \sum_{i=1}^r \lambda_i c_i(x)$$

a uvažujeme příslušnou duální úlohu

$$\begin{cases} \text{maximalizovat } \theta(\lambda), & \lambda \in \mathbb{R}^r, \\ \text{za podmínek } & \lambda_i \geq 0, i = 1, \dots, r, \end{cases} \quad (2.4)$$

kde  $\theta(\lambda) = \inf_{x \in \mathbb{R}^n} L(x; \lambda)$  je duální funkce.

Nechť  $x^* \in X^*$  je řešením úlohy (2.2). Díky předpokladu (C) lze použít KKT podmínky (viz definice 1.8), tudíž existuje vektor  $\lambda^* \in \mathbb{R}^r$  takový, že platí

$$\begin{aligned} \nabla_x L(x^*; \lambda^*) &= 0, \\ \lambda_i^* &\geq 0, \\ c_i(x^*) &\geq 0, \\ \lambda_i^* c_i(x^*) &= 0, \quad \forall i = 1, \dots, r. \end{aligned} \quad (2.5)$$



## 2.2 Funkce nelineárního rescalingu

Nyní zavedeme funkce sloužící k nelineárnímu rescalingu úlohy (2.2).

**Definice 2.2.** Dvakrát spojitě diferencovatelná funkce  $\psi : (t_0; +\infty) \rightarrow \mathbb{R}$ , kde  $-\infty < t_0 < 0$ , která splňuje podmínky

- (i)  $\psi(0) = 0, \psi'(0) = 1,$
- (ii)  $\psi'(t) > 0, \forall t \in (t_0; +\infty),$
- (iii)  $\psi''(t) < 0, \forall t \in (t_0; +\infty),$
- (iv)  $\exists a > 0 : \psi(t) \leq -at^2, \forall t \in (t_0; 0),$
- (v)  $\exists b > 0 : \psi'(t) \leq bt^{-1}, \forall t > 0,$
- (vi)  $\exists c > 0 : \psi''(t) \geq -ct^{-2}, \forall t > 0$

se nazývá **funkce nelineárního rescalingu**. Množinu těchto funkcí označíme  $\Psi$ .

**Poznámka 2.3.** Z podmínek (ii) a (iii) je patrné, že funkce nelineárního rescalingu  $\psi$  je rostoucí a konkávní na celém svém definičním oboru. Díky (v) funkce  $\psi$  rovněž splňuje podmínku  $\lim_{t \rightarrow +\infty} \psi'(t) = 0$ .

**Příklad 2.1.** Ukažte, že funkce  $\psi_1(t) = 1 - e^{-t}$  (viz obr. 1a) je funkcí nelineárního rescalingu.

*Řešení:* Funkce  $\psi_1$  je hladká, tudíž můžeme určit její první a druhou derivaci. Poté ověříme platnost podmínek z definice 2.2.

Derivace funkce  $\psi_1$  jsou ve tvaru

$$\begin{aligned}\psi_1'(t) &= e^{-t}, \\ \psi_1''(t) &= -e^{-t}.\end{aligned}$$

Z definice exponenciální funkce je zřejmé, že podmínky (i)–(iii) platí. Abychom ukázali platnost zbývajících podmínek, rozvedeme exponenciální funkci do Mac-laurinovy řady, tedy

$$e^t = \sum_{n=0}^{\infty} \frac{t^n}{n!}.$$

Odtud vidíme, že pro libovolné  $t > 0$  můžeme psát nerovnosti

$$(a) \quad e^t \geq 1 + \frac{t^2}{2}, \text{ odtud } 1 - e^t \leq -\frac{1}{2}t^2,$$

$$(b) \quad e^t \geq t, \text{ z čehož } e^{-t} \leq t^{-1},$$

$$(c) \quad e^t \geq \frac{t^2}{2}, \text{ a tudíž platí } -e^{-t} \geq -2t^{-2}.$$

Z (b), (c) je ihned vidět, že jsou splněny podmínky (v) a (vi), přičemž koeficienty  $b, c$  mají postupně hodnoty jedna a dvě. Zvolme libovolné  $t_0 \in (-\infty; 0)$ . Potom z (a) dosazením za  $t$  hodnotu  $-t$  získáváme platnost podmínky (iv),  $\forall t \in (t_0; 0)$  s koeficientem  $a = \frac{1}{2}$ .

Tímto jsme ukázali, že funkce  $\psi_1$  je skutečně funkcí nelineárního rescalingu. ■

**Poznámka 2.4.** Dalšími příklady funkcí nelineárního rescalingu jsou funkce

$$\begin{aligned} \psi_2(t) &= \ln(t+1), & \text{viz obr. 1c,} \\ \psi_3(t) &= \frac{t}{t+1}, & \text{viz obr. 1e.} \end{aligned}$$

Funkce  $\psi_i, i = 1, 2, 3$ , lze modifikovat tak, aby byly nejen definované na celém oboru reálných čísel, nýbrž i třídy  $\mathcal{C}^2(\mathbb{R})$ . Pro daný parametr  $\tau \in (-1; 0)$  se provede kvadratická extrapolace daná předpisem

$$\psi_{q_i}(t) = \begin{cases} \psi_i(t), & \text{pro } t \geq \tau, \\ q_i(t) = a_i t^2 + b_i t + c_i, & \text{pro } t < \tau, \end{cases}$$

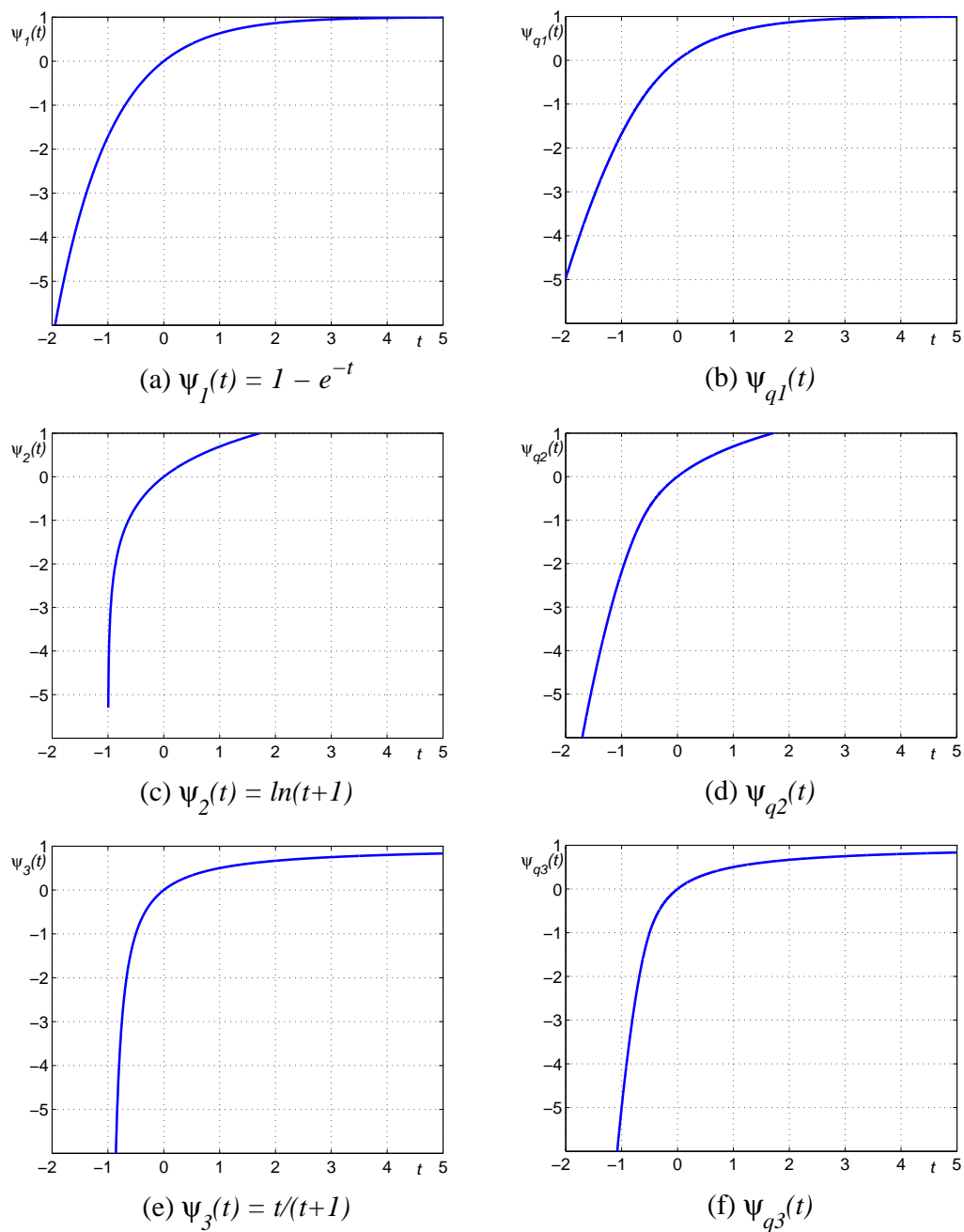
přičemž koeficienty kvadratického polynomu  $q_i$  lze určit ze vztahů

$$\psi_i(\tau) = q_i(\tau), \quad \psi'_i(\tau) = q'_i(\tau), \quad \psi''_i(\tau) = q''_i(\tau).$$

Z těchto rovností dostaneme

$$\begin{aligned} a_i &= \frac{1}{2}\psi''_i(\tau), \\ b_i &= \psi'_i(\tau) - \tau\psi''_i(\tau), \\ c_i &= \psi_i(\tau) - \tau\psi'_i(\tau) + \frac{1}{2}\tau^2\psi''_i(\tau). \end{aligned}$$

Další příklady funkcí nelineárního rescalingu s podobnými vlastnostmi lze nalézt v [10] a [15].



Obrázek 1: Grafy funkcí nelineárního rescalingu,  $\tau = -\frac{1}{2}$ .

**Poznámka 2.5.** Ve všech výpočtech v této práci použijeme jako funkci nelineárního rescalingu modifikovanou logaritmickou funkci  $\psi_{q_2}$ . Jak bylo ukázáno například ve [2], pro dostatečně širokou třídu funkcí  $c(x)$  je funkce  $-\ln c(x)$

sebeomezující funkcí. Pokud pracujeme se sebeomezujícími funkcemi, můžeme o konvergenci a volbě počátečního odhadu Newtonovy metody, kterou budeme v pozdějších úvahách používat, říci více než v obecném případě.

### 2.3 Ekvivalentní problém

Metoda NR je založena na myšlence převést pomocí jedné z funkcí  $\psi \in \Psi$  uvedených v předchozí kapitole náš problém (2.2) na úlohu

$$\begin{cases} \text{minimalizovat} & f(x), \quad x \in \mathbb{R}^n, \\ \text{za podmínku} & k^{-1}\psi(kc_i(x)) \geq 0, \quad i = 1, \dots, r, \end{cases} \quad (2.6)$$

jež je s ním ekvivalentní. Ekvivalence je chápána ve smyslu rovnosti přípustných množin a množin řešení obou úloh. Tato skutečnost je zřejmá ze zadání jednotlivých úloh a z definice funkce nelineárního rescalingu. Kladné reálné číslo  $k$  je takzvaný **parametr rescalingu**, který prozatím budeme brát jako pevně daný.

Lagrangeova funkce ekvivalentní úlohy (2.6) má tvar

$$\mathcal{L}(x; \lambda, k) = f(x) - k^{-1} \sum_{i=1}^r \lambda_i \psi_i(kc_i(x)). \quad (2.7)$$

**Lemma 2.1.** *Pro libovolné  $k \in \mathbb{R}_{++}$  a dvojici  $(x^*, \lambda^*) \in \mathbb{R}^n \times \mathbb{R}_+^r$ , pro niž jsou splněny KKT podmínky, platí*

1.  $\mathcal{L}(x^*; \lambda^*, k) = L(x^*; \lambda^*) = f(x^*)$ ,
2.  $\nabla_x \mathcal{L}(x^*; \lambda^*, k) = \nabla_x L(x^*; \lambda^*) = 0$ ,

kde  $L(x; \lambda)$  je Lagrangeova funkce původní úlohy (2.2).

**Důkaz:**

1. Platí rovnost  $\lambda_i^* c_i(x^*) = k^{-1} \lambda_i^* \psi(kc_i(x^*))$ ,  $\forall i = 1, \dots, r$ . Vskutku, pro  $\lambda_i^* = 0$  je rovnost zřejmá. Jestliže  $\lambda_i^* > 0$ , pak z komplementarity plyne, že  $c_i(x^*) = 0$ , a tedy  $\psi(kc_i(x^*)) = \psi(0) = 0$ , což je vidět z podmínky (i) pro funkci nelineárního

rescalingu. Dále použijeme definice Lagrangeových funkcí  $\mathcal{L}$  a  $L$ , dostáváme tak rovnosti

$$\begin{aligned}\mathcal{L}(x^*; \lambda^*, k) &= f(x^*) - k^{-1} \sum_{i=1}^r \lambda_i^* \psi(kc_i(x^*)) = \\ &= f(x^*) - \sum_{i=1}^r \lambda_i^* c_i(x^*) = L(x^*; \lambda^*).\end{aligned}$$

Navíc z komplementarity víme, že

$$\sum_{i=1}^r \lambda_i^* c_i(x^*) = 0,$$

tudíž platí tvrzení první části věty.

2. Zde nejdříve dokážeme, že  $\lambda_i^* = \lambda_i^* \psi'(kc_i(x^*))$ . Pro  $\lambda_i^* = 0$  je rovnost opět zřejmá. Pokud  $\lambda_i^* > 0$ , poté z komplementarity dostáváme, že  $c_i(x^*) = 0$ , a tak  $\psi'(kc_i(x^*)) = \psi'(0) = 1$  (opět z podmínky (i) pro funkci nelineárního rescalingu).

Díky této skutečnosti platí následující rovnosti

$$\begin{aligned}\nabla_x \mathcal{L}(x^*; \lambda^*, k) &= \nabla f(x^*) - \sum_{i=1}^r \lambda_i^* \psi'(kc_i(x^*)) \nabla c_i(x^*) = \\ &= \nabla f(x^*) - \sum_{i=1}^r \lambda_i^* \nabla c_i(x^*) = \\ &= \nabla_x L(x^*; \lambda^*) = 0,\end{aligned}$$

přičemž poslední rovnost plyne z KKT podmínky minima.

□

**Lemma 2.2.** *Pro libovolnou pevně zvolenou dvojici  $(\lambda, k) \in \mathbb{R}_+^r \times \mathbb{R}_{++}$  je Hessova matice Lagrangeovy funkce (2.7) pozitivně semidefinitní, což znamená, že funkce  $\mathcal{L}(x; \lambda, k)$  je konvexní v proměnné  $x$  pro každé pevně zvolené  $\lambda \in \mathbb{R}_+^r$  a parametr  $k \in \mathbb{R}_{++}$ .*

**Důkaz:** Hessova matice Lagrangeovy funkce při derivaci podle proměnné  $x$  má tvar

$$\begin{aligned}\nabla_{xx}^2 \mathcal{L}(x; \lambda, k) &= \nabla^2 f(x) - \sum_{i=1}^r (k \lambda_i \psi''(kc_i(x)) \nabla c_i(x) \nabla c_i(x)^T + \\ &\quad + \lambda_i \psi'(kc_i(x)) \nabla^2 c_i(x)).\end{aligned}$$

Nechť dvojice  $(\lambda, k) \in \mathbb{R}_+^r \times \mathbb{R}_{++}$  je libovolná, ale pevně daná. Zvolme libovolně  $p \in \mathbb{R}^n$ , pak platí

$$\begin{aligned}p^T \nabla_{xx}^2 \mathcal{L}(x; \lambda, k) p &= p^T \nabla^2 f(x) p - \sum_{i=1}^r (k \lambda_i \psi''(kc_i(x)) p^T \nabla c_i(x) \nabla c_i(x)^T p + \\ &\quad + \lambda_i \psi'(kc_i(x)) p^T \nabla^2 c_i(x)) \geq 0,\end{aligned}$$

přičemž poslední nerovnost je splněna z následujících důvodů:

1. Funkce  $f$  je konvexní, což znamená, že Hessova matice této funkce je pozitivně semidefinitní, a tedy platí  $p^T \nabla^2 f(x) p \geq 0$ .
2. Výraz  $p^T \nabla c_i(x) \nabla c_i(x)^T p$  je druhou mocninou. Vskutku, lze jej totiž přepsat do tvaru

$$(\nabla c_i(x)^T p)^T \nabla c_i(x)^T p = (\nabla c_i(x)^T p)^2,$$

tudíž je vždy nezáporný.

3. Vzhledem k tomu, že  $k\lambda_i \geq 0$  a s přihlédnutím k vlastnosti (iii) funkce nelineárního rescalingu je

$$k\lambda_i \psi''(kc_i(x)) p^T \nabla c_i(x) \nabla c_i(x)^T p \leq 0, \quad \forall i = 1, \dots, r.$$

4. Funkce  $c_i$  jsou konkávní, odtud je zřejmé, že  $p^T \nabla^2 c_i(x) p \leq 0, \forall i = 1, \dots, r$ .
5. Konečně, díky nezápornosti  $\lambda_i$  a vlastnosti (ii) funkce nelineárního rescalingu platí nerovnost

$$\lambda_i \psi'(kc_i(x)) p^T \nabla^2 c_i(x) p \leq 0, \forall i = 1, \dots, r.$$

Tímto je dokázáno tvrzení lemmatu.

□

**Důsledek 2.1.** Uvažujme situaci z lemmatu 2.2. Pokud je funkce  $f$  ryze konvexní nebo je alespoň jedna z funkcí  $c_i, i = 1, \dots, r$ , ryze konkávní, pak pro každé pevně zvolené  $\lambda \in \mathbb{R}_{++}^r$  a parametr  $k \in \mathbb{R}_{++}$  je funkce  $\mathcal{L}(x; \lambda, k)$  ryze konvexní v proměnné  $x$ .

**Důkaz:** Plyne přímo z důkazu lemmatu 2.2.

**Věta 2.1.** Pro každou dvojici  $(\lambda, k) \in \mathbb{R}_{++}^r \times \mathbb{R}_{++}$  existuje alespoň jeden bod  $\hat{x} \in \mathbb{R}^n$  takový, že

$$\mathcal{L}(\hat{x}; \lambda, k) = \min_{x \in \mathbb{R}^n} \mathcal{L}(x; \lambda, k).$$

**Důkaz:** viz [10] str. 206.

## 2.4 Metoda nelineárního rescalingu

Zamysleme se nyní nad tím, jak by mohla vypadat metoda sloužící k řešení úlohy (2.6). Pokud bychom znali řešení duální úlohy  $\lambda^* \in \mathbb{R}_+^r$ , stačilo by minimalizovat funkci  $\mathcal{L}(x; \lambda^*, k)$  v proměnné  $x$ . Úloha podmíněné optimalizace by tak byla převedena na úlohu optimalizace nepodmíněné.

Vzhledem k tomu, že optimální vektor lagrangeových multiplikátorů  $\lambda^*$  obecně neznáme, budeme se muset spokojit s jeho odhadem, který se budeme v každém kroku metody – stejně jako odhad řešení primární úlohy – snažit vylepšit. V důsledku tak přejdeme od úlohy podmíněné optimalizace k sérii úloh nepodmíněné optimalizace.

Poněvadž pro hledané  $(x^*, \lambda^*)$  jsou splněny rovnosti

$$\nabla_x \mathcal{L}(x^*; \lambda^*, k) = \nabla_x L(x^*; \lambda^*) = 0,$$

viz lemma 2.1, provede se, při známé hodnotě  $\lambda^s$  aktualizace vektoru multiplikátorů tak, aby platilo

$$\nabla_x \mathcal{L}(x^{s+1}; \lambda^s, k) = \nabla_x L(x^{s+1}; \lambda^{s+1}) = 0. \quad (2.8)$$

Díky větě 2.1 je následující **metoda nelineárního rescalingu** dobře definovaná.

**Algoritmus 2.1.** (Metoda NR)

Nechť  $k > 0$  je pevně daný parametr rescalingu,  $x^0 \in \mathbb{R}^n$  je počáteční odhad řešení a  $\lambda^0 \in \mathbb{R}_{++}^r$  je počáteční hodnota multiplikátorů. Jestliže známe hodnoty  $(x^s, \lambda^s) \in \mathbb{R}^n \times \mathbb{R}_{++}^r$ ,  $s \in \mathbb{N}_0$ , najdeme další aproximaci řešení, dvojici  $(x^{s+1}, \lambda^{s+1})$ , pomocí vztahů

$$\begin{aligned} x^{s+1} &: \nabla_x \mathcal{L}(x^{s+1}; \lambda^s, k) = 0, \\ \lambda_i^{s+1} &= \psi'(kc_i(x^{s+1})) \lambda_i^s, \quad i = 1, \dots, r. \end{aligned} \quad (2.9)$$

**Poznámka 2.6.** Jestliže  $\lambda^s \in \mathbb{R}_{++}^r$ , pak díky vlastnosti (ii) funkce  $\psi$  je zřejmé, že rovněž  $\lambda^{s+1} \in \mathbb{R}_{++}^r$ . Jinými slovy metoda NR je metodou vnitřních bodů pro duální úlohu.

**Poznámka 2.7.** Vztah pro výpočet  $\lambda^{s+1}$  ve (2.9) lze pro zjednodušení zapsat maticově ve tvaru

$$\lambda^{s+1} = \Psi'(kc(x^{s+1})) \lambda^s, \quad (2.10)$$

kde  $\Psi'(kc(x^{s+1})) = \text{diag}(\psi'(kc_i(x^{s+1})))_{i=1}^r$ .

Když se podíváme na vztahy (2.9), výpočet vektoru multiplikátorů  $\lambda^{s+1}$  je zřejmý. Otázkou ovšem zůstává, jak vyřešit soustavu nelineárních rovnic pro neznámou  $x^{s+1}$ . K tomuto účelu použijeme Newtonovu metodu (viz kapitola 1.2). Jak prakticky používat metodu NR je uvedeno v kapitole 2.6.

## 2.5 Primárně-duální metoda nelineárního rescalingu

Pro pevně dané  $s \in \mathbb{N}_0$ , parametr  $k \in \mathbb{R}_{++}$  a dvojici  $(x^s, \lambda^s) \in \mathbb{R}^n \times \mathbb{R}_{++}^r$  řešíme **primárně-duální systém** tvaru

$$\begin{aligned} \nabla_x L(x^{s+1}; \lambda^{s+1}) &= 0, \\ \lambda^{s+1} - \Psi'(kc(x^{s+1})) \lambda^s &= 0. \end{aligned} \quad (2.11)$$

Uvažíme-li rovnosti (2.8) a (2.10), je zřejmé, že se jedná o stejné vztahy jako ve (2.9).

Primárně-duální systém budeme řešit Newtonovou metodou, přičemž dvojici  $(x^s, \lambda^s)$  použijeme jako počáteční odhad.

Popíšeme si nyní jeden krok Newtonovy metody. Nechť jsou dány odhady  $x \in \mathbb{R}^n$ ,  $\lambda \in \mathbb{R}_{++}^r$ . Předpokládejme, že

$$\begin{aligned} \hat{x} &= x + \Delta x, \\ \hat{\lambda} &= \bar{\lambda} + \Delta \lambda, \end{aligned} \quad (2.12)$$

kde  $(\Delta x, \Delta \lambda)$  je **primárně-duální Newtonův krok** a  $\bar{\lambda}$  je **prediktor** vektoru  $\lambda$ , jenž vypočteme ze vztahu

$$\bar{\lambda} = \Psi'(kc(x)) \lambda,$$

který vychází z rovnosti  $\nabla_x \mathcal{L}(x; \lambda, k) = \nabla_x L(x; \bar{\lambda})$ .



Jeden krok Newtonovy metody spočívá ve vyřešení soustavy  $(n+r)$  lineárních rovnic

$$\begin{aligned} \nabla_{xx}^2 L(x; \bar{\lambda}) \Delta x - \nabla c(x)^T \Delta \lambda &= -\nabla_x L(x; \bar{\lambda}), \\ -k \Psi''(kc(x)) \Lambda \nabla c(x) \Delta x + \Delta \lambda &= 0, \end{aligned} \quad (2.13)$$

kde neznámou je dvojice  $(\Delta x, \Delta \lambda)$ . Symboly  $\Psi''(kc(x))$ ,  $\Lambda$  označují diagonální matice definované vztahy

$$\begin{aligned} \Psi''(kc(x)) &= \text{diag}(\psi''(kc_i(x)))_{i=1}^r, \\ \Lambda &= \text{diag}(\lambda_i)_{i=1}^r. \end{aligned}$$

Rovnice (2.13) lze rovněž zapsat v maticovém tvaru takto:

$$\begin{bmatrix} \nabla_{xx}^2 L(x; \bar{\lambda}) & -\nabla c(x)^T \\ -k \Psi''(kc(x)) \Lambda \nabla c(x) & \mathbf{I}_r \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} -\nabla_x L(x; \bar{\lambda}) \\ 0 \end{bmatrix}. \quad (2.14)$$

Matici soustavy označme  $N(\cdot) = N(x, \bar{\lambda}, \lambda, k)$ . Na řešení soustavy (2.14) se většinou používají metody řešení lineárních soustav s řídkou maticí (viz [16]), neboť z definice matice  $N(\cdot)$  je patrné, že by mohla obsahovat velké množství nulových prvků. Tuto skutečnost je ovšem lepší posuzovat u každé úlohy zvlášť.

Pokud ve vztazích (2.13) vyjádříme z druhé rovnosti  $\Delta \lambda$  a dosadíme do rovnosti první, dostaneme již pouze soustavu  $n$  rovnic o  $n$  neznámých ve tvaru

$$M(\cdot) \Delta x = -\nabla_x L(x; \bar{\lambda}), \quad (2.15)$$

kde

$$M(\cdot) = M(x, \bar{\lambda}, \lambda, k) = \nabla_{xx}^2 L(x; \bar{\lambda}) - k \nabla c(x)^T \Psi''(kc(x)) \Lambda \nabla c(x).$$

**Poznámka 2.8.** Je zřejmé, že matice  $M(\cdot)$  je symetrická. Podobným postupem jako v důkazu lemmatu 2.2 lze ukázat, že je matice  $M(\cdot)$  pozitivně semidefinitní. Navíc lze dokázat (viz [12] str. 120), že za předpokladu platnosti postačujících podmínek optimality druhého řádu v bodě  $(x^*; \lambda^*)$  je pro dostatečně velké  $k > 0$  matice  $M(\cdot)$  pozitivně definitní pro  $(x; \lambda)$ , které je dostatečně blízko bodu  $(x^*; \lambda^*)$ .

Vypočteme-li Newtonův krok  $\Delta x$  ze soustavy (2.15), Newtonův krok pro duální vektor určíme ze vztahu

$$\Delta \lambda = k \Psi''(kc(x)) \Lambda \nabla c(x) \Delta x. \quad (2.16)$$

Nabízí se otázka, zda je výhodnější počítat krok  $\Delta x$  ze soustavy (2.14), nebo ze vztahu (2.15). Toto dilema musíme vyřešit u každého konkrétního problému zvlášť.

Zajímavým faktem, jehož si můžeme povšimnout, je splynutí primárního Newtonova kroku s Newtonovým krokem pro nepodmíněnou minimalizaci Lagrangeovy funkce  $\mathcal{L}(x; \lambda, k)$ , o čemž hovoří následující věta.

**Věta 2.2.** *Primární Newtonův krok  $\Delta x$  ze vztahu (2.14) nebo (2.15) se shoduje s Newtonovým krokem pro nepodmíněnou minimalizaci funkce  $\mathcal{L}(x; \lambda, k)$  při pevně zvoleném  $\lambda \in \mathbb{R}_{++}^r$ ,  $k > 0$ .*

**Důkaz:** viz [14] str. 9.

Nyní shrneme poznatky této kapitoly a získáme tak algoritmus **primárně-duální metody nelineárního rescalingu** (PDNR).

**Algoritmus 2.2.** (Jeden krok PDNR metody)

Zvolme  $k \in \mathbb{R}_{++}$ . Nechť jsou dány odhady  $x \in \mathbb{R}^n$ ,  $\lambda \in \mathbb{R}_{++}^r$ . Postupujeme následovně:

1. Určíme duální prediktor

$$\bar{\lambda} = \Psi'(kc(x)) \lambda.$$

2. Vypočítáme Newtonův krok  $(\Delta x, \Delta \lambda)$  ze soustavy (2.14), nebo
- 2.' zjistíme primární Newtonův krok  $\Delta x$  ze vztahu (2.15) a najdeme duální Newtonův krok pomocí rovnosti

$$\Delta \lambda = k \Psi''(kc(x)) \Lambda \nabla c(x) \Delta x.$$

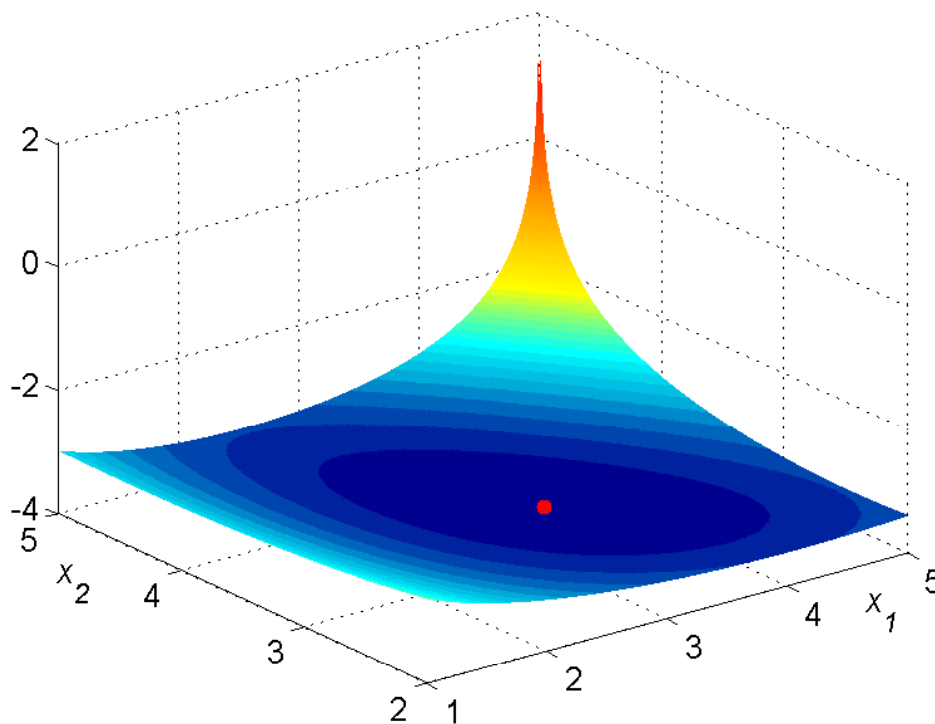
3. Vypočítáme nový primárně-duální vektor

$$x := x + \Delta x, \quad \lambda := \bar{\lambda} + \Delta \lambda.$$

**Příklad 2.2.** Vyřešte pomocí PDNR metody úlohu

$$\left\{ \begin{array}{l} \text{minimalizovat} \quad -\sum_{i=1}^3 \ln x_i, \quad x \in \mathbb{R}^3, \\ \text{za podmínek} \quad x_1 - 1 \geq 0, \\ \quad \quad \quad x_2 - 2 \geq 0, \\ \quad \quad \quad x_3 - 3 \geq 0, \\ \quad \quad \quad \sum_{i=1}^3 x_i = 10. \end{array} \right. \quad (2.17)$$

*Řešení:* Jedná se o úlohu konvexního programování, neboť účelová funkce je konvexní, podmínky ve tvaru nerovnosti jsou konkávní a podmínka ve tvaru rovnosti je lineární.

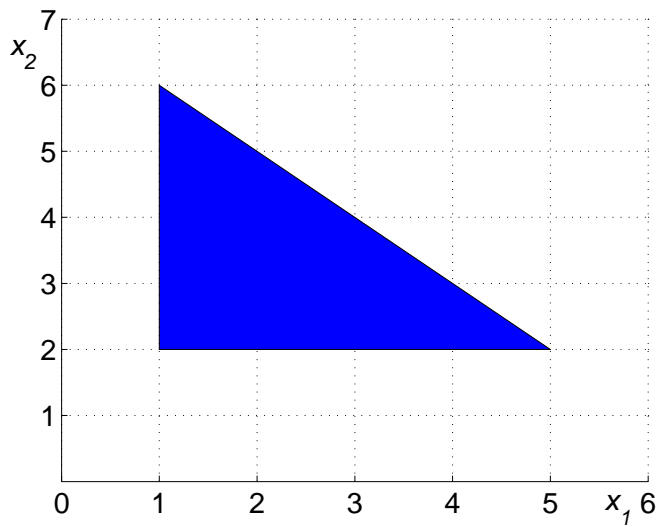


Obrázek 2: Graf funkce  $f(x_1, x_2) = -\ln x_1 - \ln x_2 - \ln(10 - x_1 - x_2)$ . Červeně je vyznačen bod minima funkce  $f$ .

Nejdříve eliminujeme podmínku ve tvaru rovnosti. Lze to udělat například tak, že se vyjádří ze vztahu  $\sum_{i=1}^3 x_i = 10$  neznámá  $x_3$ , následně se toto vyjádření

dosadí do účelové funkce a zbývajících podmínek za neznámou  $x_3$ . V důsledku touto eliminací dostáváme úlohu

$$\left\{ \begin{array}{l} \text{minimalizovat} \quad -\ln x_1 - \ln x_2 - \ln(10 - x_1 - x_2), \quad x \in \mathbb{R}^2, \\ \text{za podmínek} \quad \quad \quad x_1 - 1 \geq 0, \\ \quad \quad \quad \quad \quad \quad x_2 - 2 \geq 0, \\ \quad \quad \quad \quad \quad \quad 7 - x_1 - x_2 \geq 0. \end{array} \right. \quad (2.18)$$



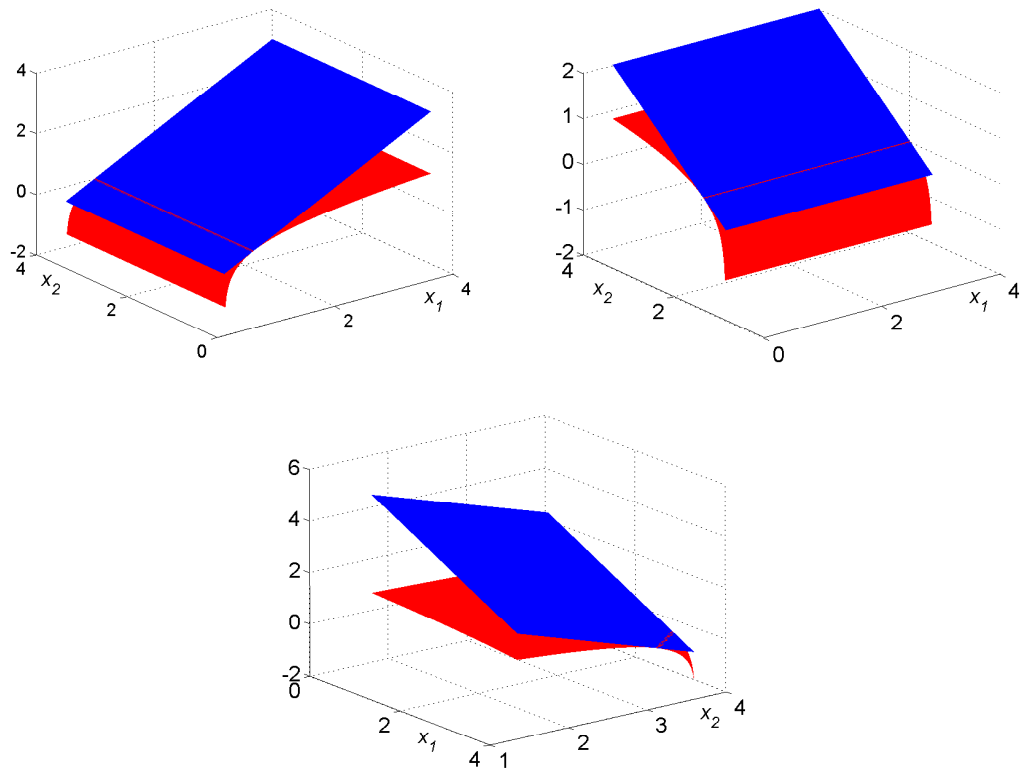
Obrázek 3: Přípustná množina úlohy (2.18).

Lagrangeova funkce úlohy (2.18) je dána předpisem

$$L(x; \lambda) = -\ln x_1 - \ln x_2 - \ln(10 - x_1 - x_2) - (\lambda_1(x_1 - 1) + \lambda_2(x_2 - 2) + \lambda_3(7 - x_1 - x_2)).$$

Převědeme problém (2.18) pomocí nelineárního rescalingu na ekvivalentní úlohu. Zvolme funkci nelineárního rescalingu  $\psi_{q_2}(t)$  a parametr  $k > 0$ , ekvivalentní úloha pak má tvar

$$\left\{ \begin{array}{l} \text{minimalizovat} \quad -\ln x_1 - \ln x_2 - \ln(10 - x_1 - x_2), \quad x \in \mathbb{R}^2, \\ \text{za podmínek} \quad \quad \quad k^{-1}\psi_{q_2}(k(x_1 - 1) + 1) \geq 0, \\ \quad \quad \quad \quad \quad \quad k^{-1}\psi_{q_2}(k(x_2 - 2) + 1) \geq 0, \\ \quad \quad \quad \quad \quad \quad k^{-1}\psi_{q_2}(k(7 - x_1 - x_2) + 1) \geq 0. \end{array} \right. \quad (2.19)$$



Obrázek 4: Srovnání grafů funkcí, jež určují přípustnou množinu v úlohách (2.18) a (2.19), před nelineárním rescalingem (modře) a po něm (červeně).

Příslušná Lagrangeova funkce úlohy 2.19 je dána předpisem

$$\begin{aligned} \mathcal{L}(x; \lambda, k) = & -\ln x_1 - \ln x_2 - \ln(10 - x_1 - x_2) - \\ & -k^{-1}(\lambda_1 \psi_{q_2}(k(x_1 - 1) + 1) + \lambda_2 \psi_{q_2}(k(x_2 - 2) + 1) + \\ & + \lambda_3 \psi_{q_2}(k(7 - x_1 - x_2) + 1)). \end{aligned}$$

Jak je patrné z obrázku 2.5, funkce nelineárního rescalingu slouží k tomu, aby podmínky ve tvaru nerovnosti převedly na podmínky, které budou:

1. Dostatečně rychle klesající pro  $x \notin S$ , kde  $S$  je přípustná množina, ve směru od hranice množiny  $S$ .
2. Dostatečně pomalu rostoucí pro  $x \in S$  ve směru od hranice množiny  $S$ .

Termíny „dostatečně rychle“ a „dostatečně pomalu“ jsou míněny ve smyslu vlastností (iv) a (v) funkce nelineárního rescalingu (viz definice 2.2).

Takovýto rescaling má za následek rychlý růst Lagrangeovy funkce  $\mathcal{L}(x; \lambda, k)$  pro  $x \notin S$  vzdalující se od množiny  $S$ , kde  $\lambda \in \mathbb{R}_{++}^r$ ,  $k \in \mathbb{R}_{++}$  jsou pevně zvolené.

Dále použijeme M-soubor `pdnr1`. Zvolíme počáteční odhad  $x^0 = (3, 3)$  a toleranci přesnosti  $\varepsilon = 10^{-4}$ ,  $\lambda^0$  a parametr  $k$  ponecháme ve výchozím nastavení. Do MATLABu postupně zadáme:

```
x = symprom(2, 'x');
x0 = [3;3];
f = - log(x(1)) - log(x(2)) - log(10-x(1)-x(2));
c(1,1) = x(1) - 1;
c(2,1) = x(2) - 2;
c(3,1) = 7 - x(1) - x(2);
[xs,fxs,s] = pdnr1(f,c,x0,x)
```

MATLAB nám odpoví:

```
xs =
    3.3333
    3.3333
fxs =
   -3.6119
s =
    3
```

Dopočítáme  $x_3^*$  tak, aby  $x_1^* + x_2^* + x_3^* = 10$ , čímž dostaneme výsledné řešení původní úlohy

$$x^* = \left( \frac{10}{3}, \frac{10}{3}, \frac{10}{3} \right).$$

Parametr  $k$  je implicitně nastaven na hodnotu  $10^4$ . Položíme-li při výpočtu  $k = 10$  resp.  $k = 1$ , dosáhneme stejného výsledku jako v předchozím případě, ovšem po čtyřech resp. dvaceti dvou krocích. ■

## 2.6 Konvergence metod NR a PDNR

V této kapitole se nejdříve zaměříme na otázku konvergence metody NR (viz algoritmus 2.1). Poté se podíváme, zda pouze přibližné řešení soustavy

$$\nabla_x \mathcal{L}(x; \lambda, k) = 0 \quad (2.20)$$

Newtonovou metodou je postačující pro konvergenci metody NR. Rovněž si ukážeme, jaké zastavovací kritérium je pro Newtonovu metodu rozumné v našem případě volit. V závěru kapitoly bude uvedena věta hovořící o konvergenci PDNR metody.

**Definice 2.3.** Řekneme, že posloupnost  $\{a_n\}_{n=0}^{+\infty} \subset \mathbb{R}^N$ , kde  $N \in \mathbb{N}$ , konverguje k bodu  $a^* \in \mathbb{R}^N$  **Q-lineárně**, jestliže

$$\limsup_{n \rightarrow +\infty} \frac{\|a_{n+1} - a^*\|}{\|a_n - a^*\|} < 1.$$

Poněvadž je limes superior dané posloupnosti reálných čísel definován jako největší hromadný bod této posloupnosti, platí následující ekvivalence.

**Poznámka 2.9.** Posloupnost  $\{a_n\}_{n=0}^{+\infty} \subset \mathbb{R}^N$ , kde  $N \in \mathbb{N}$ , konverguje k bodu  $a^* \in \mathbb{R}^N$  Q-lineárně právě tehdy, když existuje konstanta  $q \in (0; 1)$  a index  $k \in \mathbb{N}$  tak, že

$$\|a_{n+1} - a^*\| \leq q \|a_n - a^*\|, \quad \forall n \geq k.$$

Pro formulaci konvergenční věty metody NR je potřeba udělat následující označení. Nechť  $x^* \in \mathbb{R}^n$  je řešení primární úlohy (2.2) a  $\lambda^* \in \mathbb{R}_+^r$  řešení příslušné duální úlohy (2.4). Označme

$$\begin{aligned} I^+(\lambda^*) &= \{i \in \{1, \dots, r\}; \lambda_i^* > 0\}, \\ I^0(\lambda^*) &= \{i \in \{1, \dots, r\}; \lambda_i^* = 0\}. \end{aligned}$$

Dále pak pro pevně zvolené  $\lambda \in \mathbb{R}_+^r$  symbolem  $\lambda_{(1)}$  označíme vektor sestavený ze všech  $\lambda_i$  takových, že  $i \in I^+(\lambda^*)$ , a podobně symbolem  $\lambda_{(2)}$  označíme vektor sestavený ze všech  $\lambda_i$ ,  $i \in I^0(\lambda^*)$ .

Zvolme  $\delta \in \mathbb{R}_{++}$  dostatečně malé tak, že platí

$$\delta < \lambda_i^*, \quad \forall i \in I^+(\lambda^*),$$

a zvolme  $k_0 \in \mathbb{R}_{++}$  dostatečně velké.

**Definice 2.4.** Definujeme **rozšířený duální definiční obor**  $D(\delta)$  předpisem

$$D(\delta) = \{(\lambda, k); (\lambda_{(1)}, k) \in D^+(\delta), (\lambda_{(2)}, k) \in D^0(\delta)\},$$

kde

$$\begin{aligned} D^+(\delta) &= \{(\lambda_{(1)}, k); \lambda_i \geq \delta, |\lambda_i - \lambda_i^*| \leq k\delta, k \geq k_0, \forall i \in I^+(\lambda^*)\}, \\ D^0(\delta) &= \{(\lambda_{(2)}, k); 0 \leq \lambda_i \leq k\delta, k \geq k_0, \forall i \in I^0(\lambda^*)\}. \end{aligned}$$

Říkáme, že  $D^+(\delta)$  je **aktivní částí** a  $D^0(\delta)$  je **pasivní částí** množiny  $D(\delta)$ .

**Věta 2.3.** *Nechť jsou splněny předpoklady (A) a (C), viz kapitola 2.1, poté pro všechna  $(\lambda, k) \in D(\delta)$  platí:*

(i) *Existuje  $\hat{x} = \hat{x}(\lambda, k)$  takové, že  $\nabla_x \mathcal{L}(\hat{x}; \lambda, k) = 0$ .*

(ii) *Pro  $(\hat{x}, \hat{\lambda})$ , vypočtených pomocí vztahů (2.9), existuje reálná konstanta  $c > 0$  nezávislá na parametru  $k$  taková, že platí následující odhady:*

$$\begin{aligned} \|\hat{x} - x^*\| &\leq ck^{-1} \|\lambda - \lambda^*\|, \\ \|\hat{\lambda} - \lambda^*\| &\leq ck^{-1} \|\lambda - \lambda^*\|. \end{aligned}$$

*Navíc platí rovnosti  $\hat{x}(\lambda^*, k) = x^*$  a  $\hat{\lambda}(\lambda^*, k) = \lambda^*$ .*

(iii) *Lagrangeova funkce  $\mathcal{L}(x; \lambda, k)$  je silně konvexní v okolí bodu  $\hat{x}(\lambda, k)$ .*

**Důkaz:** viz [14] str. 43.

**Poznámka 2.10.** Díky větě 2.3 můžeme nahlédnout, že formulace „dostatečně velký parametr  $k_0$ “ nám říká volit  $k_0$  aspoň takové, aby  $ck_0^{-1} < 1$ .



**Důsledek 2.2.** Z předchozí věty přímo plyne Q-lineární rychlost konvergence primárně-duální posloupnosti  $\{(x^s, \lambda^s)\}$  generované vztahy 2.9, neboť jestliže  $(\lambda^s, k) \in D(\delta)$  pro nějaké  $s \in \mathbb{N}_0$ , pak i  $(\lambda^{s+1}, k) \in D(\delta)$ .

Navíc, pokud označíme  $q = ck^{-1}$ , můžeme pro libovolné  $q \in (0; 1)$  najít  $k_q \geq k_0$  takové, že pro libovolné  $k \geq k_q$  platí odhady:

$$\begin{aligned}\|x^{s+1} - x^*\| &\leq q\|\lambda^s - \lambda^*\|, \\ \|\lambda^{s+1} - \lambda^*\| &\leq q\|\lambda^s - \lambda^*\|.\end{aligned}$$

Jak již bylo řečeno, každý krok metody NR obnáší vyřešit úlohu (2.20). Ovšem Newtonova metoda, za pomoci které chceme tuto úlohu řešit, je metodou infinitní. Což znamená, že po konečném počtu kroků Newtonovy metody v obecném případě nenajdeme přesné řešení úlohy (2.20), ale pouze jeho aproximaci. Na otázku, zda metoda NR bude konvergovat i přes toto úskalí, odpovídá následující věta.

**Věta 2.4.** *Nechť platí předpoklady (A) a (C) a nechť  $(\lambda, k) \in D(\delta)$ . Předpokládejme dále, že Hessovy matice  $\nabla^2 f$  a  $\nabla^2 c_i$ ,  $i = 1, \dots, r$ , splňují Lipschitzovu podmínku, tj.*

$$\begin{aligned}\|\nabla^2 f(x_1) - \nabla^2 f(x_2)\| &\leq L_0\|x_1 - x_2\|, \quad \forall x_1, x_2 \in \mathcal{D}(f), \\ \|\nabla^2 c_i(x_1) - \nabla^2 c_i(x_2)\| &\leq L_i\|x_1 - x_2\|, \quad \forall x_1, x_2 \in \mathcal{D}(c_i), \quad \forall i = 1, \dots, r.\end{aligned}\tag{2.21}$$

Označme  $\tilde{x}$  přibližné řešení soustavy rovnic  $\nabla_x \mathcal{L}(x, \lambda, k) = 0$  takové, že platí nerovnost

$$\|\nabla_x \mathcal{L}(\tilde{x}; \lambda, k)\| \leq \sigma k^{-1} \|\tilde{\lambda} - \lambda\|,$$

kde  $\sigma \in \mathbb{R}_{++}$  a  $\tilde{\lambda} = \Psi'(kc(\tilde{x}))\lambda$ . Pak pro dvojici  $(\tilde{x}, \tilde{\lambda})$  platí následující odhady:

$$\begin{aligned}\|\tilde{x} - x^*\| &\leq c(1 + \sigma)k^{-1}\|\lambda - \lambda^*\|, \\ \|\tilde{\lambda} - \lambda^*\| &\leq c(1 + \sigma)k^{-1}\|\lambda - \lambda^*\|,\end{aligned}\tag{2.22}$$

kde  $c \in \mathbb{R}_{++}$  nezávisí na  $k$ .

**Důkaz:** Provede se podobně jako důkaz analogického tvrzení v [9] str. 445.

**Důsledek 2.3.** Mějme posloupnost  $\{(\tilde{x}^s, \tilde{\lambda}^s)\}$ , pro kterou platí

$$\begin{aligned}\|\nabla_x \mathcal{L}(\tilde{x}^{s+1}; \tilde{\lambda}^s, k)\| &\leq \sigma k^{-1} \|\tilde{\lambda}^{s+1} - \tilde{\lambda}^s\|, \\ \tilde{\lambda}^{s+1} &= \Psi'(kc(\tilde{x}^{s+1}))\tilde{\lambda}^s, \quad \forall s \in \mathbb{N}_0.\end{aligned}\tag{2.23}$$

Z předchozí věty plyne, že posloupnost  $\{(\tilde{x}^s, \tilde{\lambda}^s)\}$ , jež je aproximací primárně-duální posloupnosti  $\{(x^s, \lambda^s)\}$ , konverguje k řešení  $(x^*, \lambda^*)$  rovněž Q-lineárně.

Navíc, pokud označíme  $q = c(1 + \sigma)k^{-1}$ , můžeme pro libovolné  $q \in (0; 1)$  najít  $k_q \geq k_0$  takové, že pro libovolné  $k \geq k_q$  platí odhad:

$$\max \left\{ \|\tilde{x}^{s+1} - x^*\|, \|\tilde{\lambda}^{s+1} - \lambda^*\| \right\} \leq q \|\tilde{\lambda}^s - \lambda^*\|.$$

Vztahů (2.23) lze využít jako zastavovacího kritéria Newtonovy metody. Nyní již tedy víme, jak prakticky používat metodu NR. Dále o ní budeme hovořit jako o **Newtonově metodě NR** nebo ji jen jednoduše označovat za metodu NR.

**Poznámka 2.11.** Při použití zastavovacího kritéria (2.23) může nalezení odhadu  $\tilde{x}^{s+1}$  vyžadovat několik kroků Newtonovy metody. Pokud od nějakého bodu je potřeba pouze jediného kroku Newtonovy metody, nazýváme tento jev „**žhavým startem**“ metody NR.

Pro dané  $\varepsilon \in \mathbb{R}_{++}$  označme symbolem  $\mathcal{U}_\varepsilon$  uzavřené epsilonové okolí bodu  $z^* = (x^*, \lambda^*)$ , tedy

$$\mathcal{U}_\varepsilon = \{z = (x, \lambda) \in \mathbb{R}^n \times \mathbb{R}_+^r; \|z - z^*\| \leq \varepsilon\}. \quad (2.24)$$

Následující věta hovoří o existenci žhavého startu v teorii NR.

**Věta 2.5.** *Jestliže platí předpoklady (A), (C) a jsou splněny Lipschitzovy podmínky (2.21), potom pro libovolné  $q \in (0; 1)$  existuje  $k_q \geq k_0$  a  $\varepsilon > 0$  dostatečně malé tak, že pro libovolné  $k \geq k_q$  a libovolnou dvojici  $(x^s, \lambda^s) \in \mathcal{U}_\varepsilon$ , kterou vezmeme za počáteční odhad v algoritmu 2.2, je potřeba pouze jednoho PDNR kroku, abychom získali novou aproximaci  $(x^{s+1}, \lambda^{s+1})$  takovou, že platí nerovnost*

$$\max \left\{ \|x^{s+1} - x^*\|, \|\lambda^{s+1} - \lambda^*\| \right\} \leq q\varepsilon.$$

**Důkaz:** viz [14] str. 141.

**Důsledek 2.4.** Díky větě 2.2 plyne z předchozí věty Q-lineární rychlost konvergence PDNR metody.

**Poznámka 2.12.** Řešíme-li úlohu konvexního programování (2.2) metodou NR, pak věty 2.2 a 2.5 říkají, že metoda NR od určitého kroku přejde v PDNR metodu.

**Příklad 2.3.** Vyřešte úlohu (2.18) pomocí metody NR.

*Řešení:* Použijeme M-soubor `nr1` a v průběhu budeme sledovat počet vnitřních Newtonových kroků metody. Zvolíme stejný počáteční odhad  $x^0 = (3, 3)$  a toleranci přesnosti  $\varepsilon = 10^{-4}$  jako v příkladě 2.2. Vektor multiplikátorů  $\lambda^0$  a parametr  $k$  ponecháme ve výchozím nastavení. Parametr  $\sigma$  ze vztahů (2.23) položíme roven jedné.

Zadání úlohy do MATLABu a jeho reakce – řešení úlohy – jsou stejné jako v již zmíněném příkladě 2.2 s tím rozdílem, že voláme M-soubor `nr1`, tedy zadáváme příkaz `[xs, fxs, s] = nr1(f, c, x0, x)`.

Opět jsou potřeba tři vnější kroky metody. Zajímavé je, že před první aktualizací hodnoty multiplikátorů se provedou dva vnitřní Newtonovy kroky. Poté metoda NR přejde v PDNR metodu. Jinými slovy, ve zbývajících dvou vnějších krocích je vždy jeden krok Newtonovy metody postačující k tomu, aby bylo splněno zastavovací kritérium (2.23). Takže v tomto případě po prvním kroku metody NR dochází ke žhavému startu. ■

### 3 Vylepšení metod založených na NR

V této sekci se budeme zabývat vylepšením metod založených na teorii NR. Jedna z vlastností, které nás budou zajímat, se týká zvýšení rychlosti konvergence PDNR metody. Takovéto otázky se věnuje kapitola 3.1.

Konvergence metod založených na NR, tak jak prozatím byly uvedeny, stojí a padá s úspěšností Newtonovy metody. Přestože je klasická Newtonova metoda velmi efektivní v okolí bodu řešení, konverguje pouze lokálně. Totéž tedy platí jednak pro metodu NR, tak i pro PDNR metodu. Nabízí se několik možností, jak se s touto skutečností vypořádat:

- volba délky kroku,
- kombinace s globálně konvergentní metodou,
- Levenberg-Marquardtova regularizace.

V kapitolách 3.2, 3.3 a 3.4 jsou výše uvedené možnosti rozebrány.

#### 3.1 Dynamická změna parametru rescalingu

Prozatím jsme pracovali s pevně daným parametrem rescalingu  $k \in \mathbb{R}_{++}$ , který jsme považovali za dostatečně velký. To nám nejen zaručí Q-lineární konvergenci metody NR, ale i umožní vyhnout se špatné podmíněnosti Hessovy matice funkce  $\mathcal{L}(x; \lambda, k)$ .

Za účelem zvýšit rychlost konvergence metody NR musíme v každém kroku metody zvyšovat hodnotu parametru  $k$  – **metoda NR s dynamickou změnou parametru rescalingu** (NRD). To pro velmi vysoké hodnoty parametru  $k$  ovšem povede k řešení úloh nepodmíněné minimalizace se špatně podmíněnou maticí  $\nabla_{xx}^2 \mathcal{L}(x; \lambda, k)$ . Z tohoto důvodu ve finální fázi použijeme PDNR metodu, neboť ta neprovádí minimalizaci, a tudíž je špatná podmíněnost nepodstatná.

Nejdříve definujme funkci, pomocí níž budeme měřit vzdálenost odhadu řešení od skutečného řešení  $(x^*, \lambda^*)$ .

**Definice 3.1.** Funkce  $\nu : \mathbb{R}^n \times \mathbb{R}^r \rightarrow \mathbb{R}_+$ , definovaná předpisem

$$\nu(x, \lambda) = \max \left\{ \|\nabla_x L(x; \lambda)\|, -\min_{1 \leq i \leq r} c_i(x), \sum_{i=1}^r |\lambda_i c_i(x)|, -\min_{1 \leq i \leq r} \lambda_i \right\}, \quad (3.1)$$

se nazývá **hodnotící funkce**.

Důvodem k vytvoření hodnotící funkce právě ve tvaru (3.1) jsou KKT nutné a postačující podmínky optimality prvního řádu (viz kapitola 1.1.2). S přihlédnutím k faktu, že se zabýváme úlohou konvexního programování, z podmínek optimality prvního řádu rovněž plyne platnost následující ekvivalence:

$$\nu(\hat{x}, \hat{\lambda}) = 0 \Leftrightarrow (\hat{x}, \hat{\lambda}) \in X^*.$$

**Poznámka 3.1.** Vzhledem k tomu, že jsou metody NR a PDNR metodami vnitřních bodů pro multiplikátory  $\lambda$ , není potřeba při výpočtu hodnoty funkce  $\nu(x, \lambda)$  uvažovat výraz „ $-\min_{1 \leq i \leq r} \lambda_i$ “, neboť platí nerovnosti

$$-\min_{1 \leq i \leq r} \lambda_i \leq 0 \leq \|\nabla_x L(x; \lambda)\|, \quad \forall (x, \lambda) \in \mathbb{R}^n \times \mathbb{R}_+^r.$$

Předpokládejme, že máme posloupnost odhadů řešení  $\{(x^s, \lambda^s)\}_{s=0}^{+\infty}$ . Hodnotu parametru rescalingu v  $s$ -tém kroku nastavíme podle vztahu

$$k_s = \nu(x^s, \lambda^s)^{-1/2}. \quad (3.2)$$

Je zřejmé, že pokud posloupnost odhadů řešení konverguje k dvojici  $(x^*, \lambda^*)$ , potom  $\nu(x^s, \lambda^s) \rightarrow 0^+$ , a tedy také  $k_s \rightarrow +\infty$ , pro  $s \rightarrow +\infty$ .

Algoritmy metody NRD a **primárně-duální metody nelineárního rescalingu s dynamickou změnou parametru** (PDNRD) jsou podobné algoritmům 2.1, 2.2 bez dynamické změny, až na příkaz měnící v každém kroku parametr rescalingu podle vztahu (3.2).

**Poznámka 3.2.** Pro metodu NRD platí analogické úvahy a tvrzení k těm uvedeným v kapitole 2.6 pro metodu NR.

Abychom předešli singularitě matice v soustavě (2.14), můžeme namísto této soustavy řešit regularizovanou soustavu ve tvaru:

$$\begin{bmatrix} \nabla_{xx}^2 L(x; \bar{\lambda}) + \frac{1}{k^2} \mathbf{I}_n & -\nabla c(x)^T \\ -k\Psi''(kc(x)) \Lambda \nabla c(x) & \mathbf{I}_r \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} -\nabla_x L(x; \bar{\lambda}) \\ 0 \end{bmatrix}, \quad (3.3)$$

kde dvojice  $(x, \lambda)$  je odhad řešení,  $k$  je aktuální hodnota parametru rescalingu a  $\bar{\lambda}$  je prediktor vektoru multiplikátorů.

Matici výše uvedené soustavy označme  $N_k(\cdot) = N_k(x, \bar{\lambda}, \lambda, k)$ . Je důležité poznamenat, že takto provedená regularizace nijak neovlivňuje rychlost konvergence PDNRD metody.

**Definice 3.2.** Řekneme, že posloupnost  $\{a_n\}_{n=0}^{+\infty} \subset \mathbb{R}^N$ , kde  $N \in \mathbb{N}$ , konverguje k bodu  $a^* \in \mathbb{R}^N$  **1,5-Q-superlineárně**, jestliže existuje konstanta  $C \in (0; +\infty)$  a index  $k \in \mathbb{N}$  takový, že

$$\|a_{n+1} - a^*\| \leq C \|a_n - a^*\|^{3/2}, \quad \forall n \geq k.$$

Z následující věty plyne 1,5-Q-superlineární rychlost konvergence PDNRD metody.

**Věta 3.1.** *Jestliže platí předpoklady (A), (C) a jsou splněny Lipschitzovy podmínky (2.21), potom existuje  $\varepsilon > 0$  dostatečně malé takové, že pro libovolnou dvojici  $z^s := (x^s, \lambda^s) \in \mathcal{U}_\varepsilon$  (viz (2.24)) stačí pouze jeden krok PDNRD metody k dosažení nového odhadu řešení  $z^{s+1} := (x^{s+1}, \lambda^{s+1})$  takového, že platí nerovnost*

$$\|z^{s+1} - z^*\| \leq C \|z^s - z^*\|^{3/2},$$

kde  $z^* := (x^*, \lambda^*)$  a konstanta  $C \in (0; +\infty)$ .

**Důkaz:** viz [13] str. 247.

## 3.2 Volba délky kroku

Široce využívanou globalizační strategií je metoda volby délky kroku. Uvažujme situaci popsanou v kapitole 1.2. Newtonova metoda pro úlohu nepodmíněné minimalizace funkce  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  s volbou délky kroku – tzv. **tlumená**

**Newtonova metoda** – vypadá následovně:

$$\begin{aligned}\nabla^2 f(x_k)\Delta x &= -\nabla f(x_k), \\ \alpha_k \in (0; 1) : \quad f(x_k + \alpha_k\Delta x) - f(x_k) &\leq \eta\alpha_k\Delta x^T\nabla f(x_k), \\ x_{k+1} &= x_k + \alpha_k\Delta x,\end{aligned}$$

kde  $x_k \in \mathbb{R}^n$  je již známý odhad řešení,  $\alpha_k \in (0; 1)$  je délka  $k$ -tého kroku a  $\eta \in (0; 1)$  je vhodně zvolený parametr. Podmínka pro určení hodnoty  $\alpha_k$  je jedna z **Wolfeho podmínek**, která bývá označována jako **Goldstein-Armijova podmínka**. Podrobně je problematika volby délky kroku popsána například v [6] str. 375.

Prakticky můžeme Goldstein-Armijovu podmínku realizovat například pomocí algoritmu **zpětného vyhledávání**. Nejedná se sice o metodu exaktní, avšak výhodou je její jednoduchost a efektivita. Algoritmus metody zpětného vyhledávání je popsán ve [2] str. 27–30. Na témže místě je rovněž uveden M-soubor této metody.

V našem případě je použití metody volby délky kroku popsáno v následujícím algoritmu.

**Algoritmus 3.1.** Nechť je dvojice  $(\lambda, k) \in \mathbb{R}_{++}^r \times \mathbb{R}_{++}$  pevně dána. Postupujeme iteračně. Předpokládejme, že odhad  $x^s \in \mathbb{R}^n$ ,  $s \in \mathbb{N}_0$ , řešení úlohy nepodmíněné minimalizace funkce  $\mathcal{L}(x; \lambda, k)$  je znám. Další aproximaci  $x^{s+1}$  nalezneme pomocí postupu:

1. Ze vztahu (3.3) určíme  $\Delta x$ .
2. Najdeme  $\alpha_s \in (0; 1)$  pomocí metody zpětného vyhledávání.
3. Položíme  $x^{s+1} = x^s + \alpha_s\Delta x$ .

Tento postup opakujeme tak dlouho, dokud není splněno zastavovací kritérium (viz vztahy (2.23)).

**Lemma 3.1.** *Nechť  $\mathcal{U}_\varepsilon$  je uzavřené epsilonové okolí bodu  $(x^*, \lambda^*)$  stejné jako ve větě 3.1. Pro libovolnou dvojici  $(x, \lambda) \notin \mathcal{U}_\varepsilon$ , kde vektor  $\lambda \in \mathbb{R}_{++}^r$  je pevně*

zvolený, a parametr  $k \in \mathbb{R}_{++}$  rovněž pevně zvolený, algoritmus 3.1 konverguje k řešení úlohy

$$\min_{x \in \mathbb{R}^n} \mathcal{L}(x; \lambda, k).$$

**Důkaz:** viz [13] str. 253.

**Věta 3.2.** *Za předpokladů věty 3.1 PDNRD metoda konverguje s asymptotickou 1,5-Q-superlineární rychlostí konvergence.*

**Důkaz:** Plyne z věty 3.1, poznámky 3.2 a lemmatu 3.1.

Jak tedy prakticky postupovat při řešení úlohy (2.2)?

Předpokládejme, že známe odhad  $(x, \lambda)$  řešení úlohy (2.2). K nalezení následujícího odhadu  $(\hat{x}, \hat{\lambda})$  nejdříve zkusíme použít PDNRD metodu. Pokud dojde k superlineárnímu poklesu hodnotící funkce, pak jsme našli následující odhad. Pokud ne, využijeme Newtonův krok  $\Delta x$  vypočtený v PDNRD metodě k minimalizaci funkce  $\mathcal{L}(x; \lambda, k)$  pro pevné  $\lambda, k$ . Při této minimalizaci, za účelem zaručit globální konvergenci k nepodmíněnému minimu funkce  $\mathcal{L}(x; \lambda, k)$ , použijeme metodu zpětného vyhledávání.

Výše popsáný postup řešení úlohy (2.2) povede úspěšně k cíli, neboť:

- mimo okolí  $\mathcal{U}_\varepsilon$  platí lemma 3.1, a tedy máme zaručenu globální konvergenci Newtonovy NR metody k okolí bodu primárně-duálního řešení  $(x^*, \lambda^*)$ ,
- platí věta 3.1, jež nám zaručuje 1,5-Q-superlineární konvergenci PDNRD metody v uzavřeném okolí  $\mathcal{U}_\varepsilon$  řešení naší úlohy.

V závěru této kapitoly shrneme všechny poznatky a uvedeme algoritmus globálně konvergentní PDNRD metody.

**Algoritmus 3.2.** (Globálně konvergentní PDNRD metoda)

Nechť je dán počáteční odhad  $x^0 \in \mathbb{R}^n$ , tolerance přesnosti  $\varepsilon > 0$  a počáteční hodnota parametru rescalingu  $k \in \mathbb{R}_{++}$ .

Zvolme parametry  $q \in (0; 1)$ ,  $\eta \in (0; 0,5)$ ,  $\omega > 1$ ,  $\sigma > 0$  a  $\theta > 0$ . Položme  $x := x^0$ ,  $\lambda := (1, 1, \dots, 1) \in \mathbb{R}^r$  a  $H := \nu(x, \lambda)$ . Postupujeme takto:



(1°) **Jestliže**  $H \leq \varepsilon$ , **pak** ukončíme výpočet, **výsledkem** je dvojice  $(x, \lambda)$ .

(2°) Určíme  $(\Delta x, \Delta \lambda)$  ze vztahu (3.3) a položíme

$$\hat{x} := x + \Delta x, \quad \hat{\lambda} := \lambda + \Delta \lambda, \quad \hat{H} := \nu(\hat{x}, \hat{\lambda}).$$

(3°) **Jestliže**  $\hat{H} \leq \min \{H^{3/2-\theta}, 1 - \theta\}$ , **pak** položíme

$$x := \hat{x}, \quad \lambda := \hat{\lambda}, \quad H := \hat{H}, \quad k := \max \left\{ \frac{1}{\sqrt{H}}, k \right\}$$

a přejdeme na krok (1°).

(4°) Určíme  $\alpha \in (0; 1)$  tak, aby platilo

$$\mathcal{L}(x + \alpha \Delta x; \lambda, k) - \mathcal{L}(x; \lambda, k) \leq \eta \alpha \Delta x^T \nabla_x \mathcal{L}(x; \lambda, k),$$

pomocí metody zpětného vyhledávání.

(5°) Položíme

$$x := x + \alpha \Delta x, \quad \hat{\lambda} := \Psi'(kc(x))\lambda.$$

(6°) **Jestliže** platí

$$\|\nabla_x \mathcal{L}(x; \lambda, k)\| \leq \frac{\sigma}{k} \|\hat{\lambda} - \lambda\|,$$

**pak** přejdeme na krok (8°).

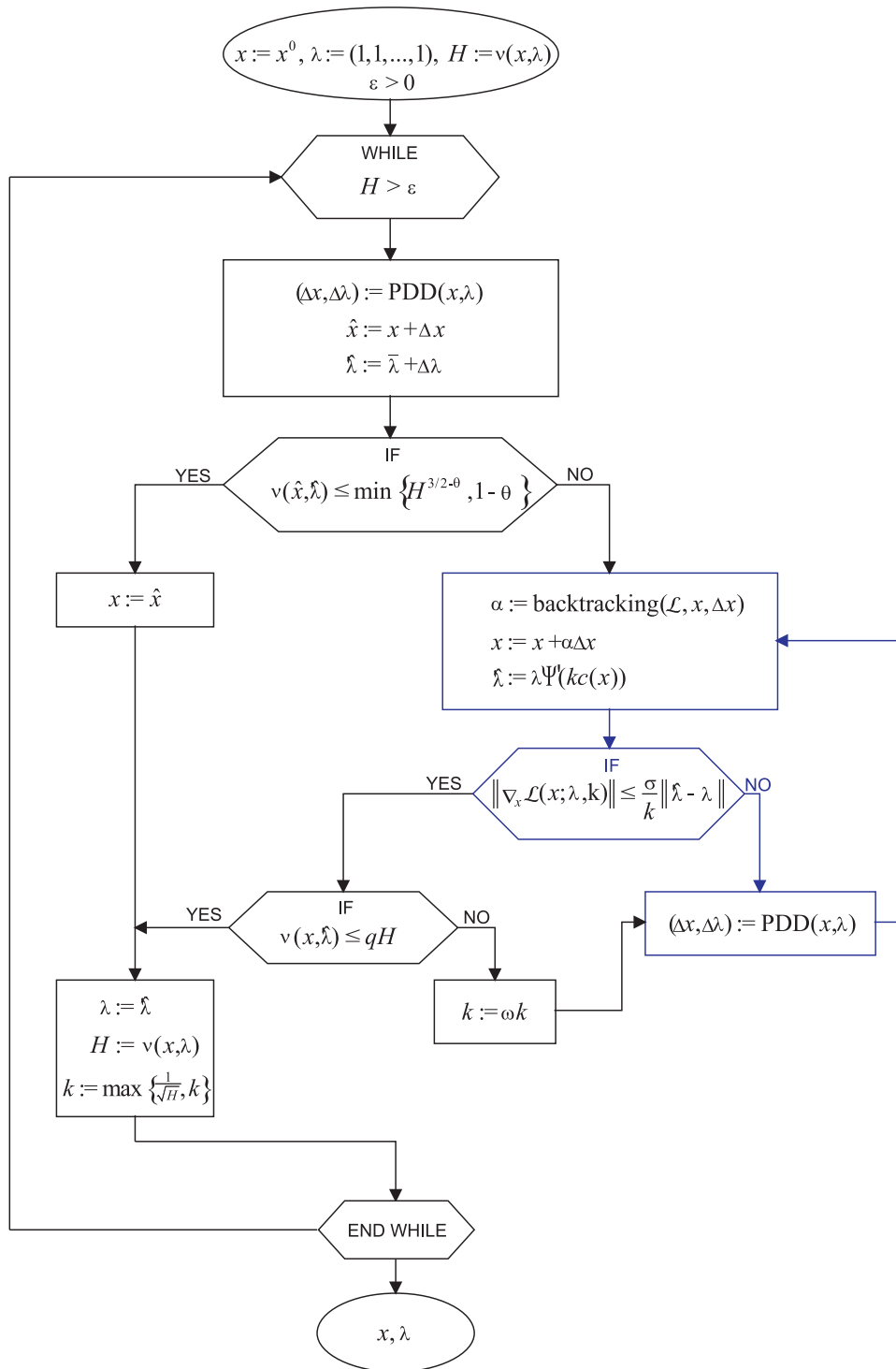
(7°) Určíme  $(\Delta x, \Delta \lambda)$  ze vztahu (3.3) a přejdeme na krok (4°).

(8°) **Jestliže**  $\nu(x, \hat{\lambda}) \leq qH$ , **pak** položíme

$$\lambda := \hat{\lambda}, \quad H := \nu(x, \hat{\lambda}), \quad k := \max \left\{ \frac{1}{\sqrt{H}}, k \right\}$$

a přejdeme na krok (1°).

(9°) Položíme  $k := \omega k$  a přejdeme na krok (7°).



Obrázek 5: Vývojový diagram globálně konvergentní PDNRD metody, kde funkce „PDD“ resp. „backtracking“ označují řešení soustavy (3.3) resp. metodu zpětného vyhledávání.

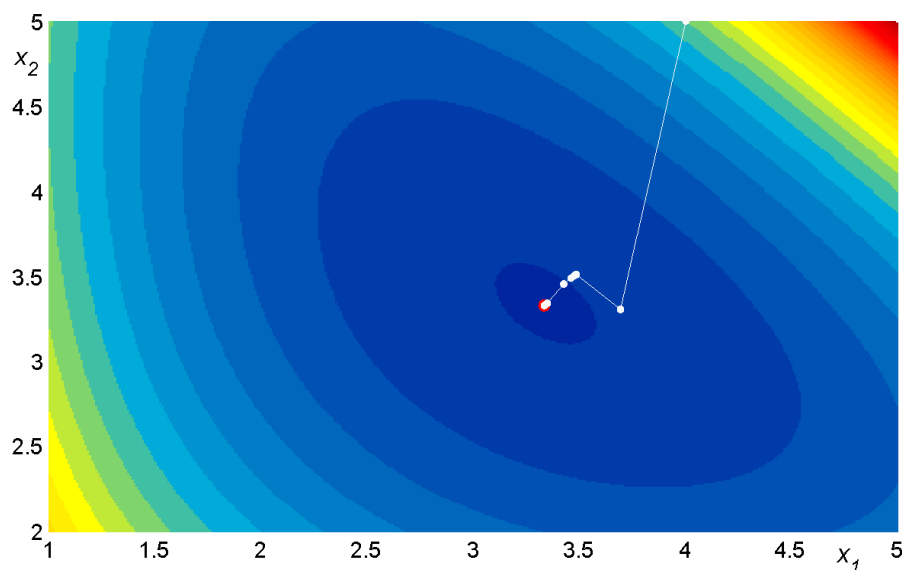
$\omega > 1$	faktor zvyšující hodnotu parametru rescalingu
$\sigma > 0$	parametr ovlivňující zastavovací kritérium tlumené Newtonovy metody
$\theta > 0$	parametr kontrolující rychlost konvergence
$q \in (0; 1)$	faktor, o který chceme v každém kroku metody zmenšit vzdálenost mezi odhadem a řešením
$\eta \in (0; 0,5)$	parametr zpětného vyhledávání

Tabulka 1: Význam parametrů globálně konvergentní PDNRD metody.

Otázka vhodné volby parametrů PDNRD metody je řešena v kapitole 4.1.

**Příklad 3.1.** Řešte úlohu (2.18) pomocí metody NR. Za počáteční odhad zvolte  $x^0 = (4, 5)$ .

*Řešení:* Počáteční odhad  $x^0$  není přípustným bodem naší úlohy, to ovšem nevadí, neboť metody založené na NR nejsou metodami vnitřních bodů. Díky tomuto je mnohdy velice snadné najít vhodný počáteční odhad.



Obrázek 6: Průřez grafu funkce  $f(x_1, x_2) = -\ln x_1 - \ln x_2 - \ln(10 - x_1 - x_2)$ . Červeně je vyznačen bod minima této funkce. Bílou barvou je znázorněn postup PDNRD metody.

Zadání úlohy do MATLABu je analogické jako v příkladě 2.2. Použijeme M-soubor `pdnrd`. Na počátku nastavíme parametr rescalingu na hodnotu sto, tudíž

PDNRD metodu zavoláme příkazem:

```
[xs, fxs, s] = pdnrd(f, c, x0, 100)
```

Výsledkem je:

```
xs =  
    3.3333  
    3.3333  
fxs =  
   -3.6119  
s =  
    4
```

Metoda našla řešení ve čtyřech krocích, přičemž hodnotící funkce nabývá v tomto bodě hodnoty menší než  $10^{-6}$ . V prvním kroku PDNRD metody proběhne šest vnitřních kroků a poté dojde ke žhavému startu. Postup metody je ilustrován na obrázku 6. ■

Abychom ukázali použitelnost PDNRD metody, vyřešíme pomocí ní další, již složitější, úlohy. Zadání a řešení úloh je uvedeno v kapitole 4.

### 3.3 Primárně-duální metoda sledování cesty

Ve své práci [12] uvedli Roman Polyak a Igor Griva algoritmus, jenž kombinuje PDNR metodu s **primárně-duální metodou sledování cesty**. Jako výsledek tak dostali globálně konvergentní metodu mající výhody obou zmíněných metod a postrádající jejich hlavní nevýhody.

#### 3.3.1 Metoda sledování cesty

Uvažujme úlohu (2.2) a k ní příslušnou duální úlohu (2.4). Pro jednoduchost zápisu označme  $e = (1, 1, \dots, 1) \in \mathbb{R}^r$ .

**Definice 3.3.** Pro danou transformaci  $\psi \in \Psi$  definujeme modifikovanou bariérovou funkci  $P : \mathbb{R}^n \times \mathbb{R}_+ \rightarrow \mathbb{R}$  předpisem

$$P(x; k) = \mathcal{L}(x; e, k) = f(x) - k^{-1} \sum_{i=1}^r \psi(kc_i(x)).$$

Díky větě 2.1 víme, že pro libovolné pevně zvolené  $k \in \mathbb{R}_{++}$  existuje bod minima funkce  $P(x; k)$ . Jinými slovy, existuje bod  $\hat{x} = \hat{x}(k)$  takový, že platí rovnosti

$$\nabla_x P(\hat{x}; k) = \nabla f(\hat{x}) - \sum_{i=1}^r \psi'(kc_i(\hat{x})) \nabla c_i(\hat{x}) = 0.$$

Označíme-li

$$\hat{\lambda}_i = \hat{\lambda}_i(k) = \psi'(kc_i(\hat{x})), \quad \forall i = 1, \dots, r,$$

potom platí rovnosti

$$\nabla P(\hat{x}; k) = \nabla f(\hat{x}) - \sum_{i=1}^r \hat{\lambda}_i \nabla c_i(\hat{x}) = \nabla_x L(\hat{x}; \hat{\lambda}) = 0.$$

Uvedeme nyní algoritmus **metody sledování cesty** (PF)<sup>1</sup>.

**Algoritmus 3.3.** (Metoda PF)

Mějme dānu rostoucí posloupnost  $\{k_s\}_{s=0}^{+\infty}$  takovou, že  $k_s \rightarrow +\infty$  pro  $s \rightarrow +\infty$ . Nechtě  $x^0 \in \mathbb{R}^n$  je počáteční odhad řešení úlohy (2.2). Jestliže znāme hodnotu  $x^s \in \mathbb{R}^n$ ,  $s \in \mathbb{N}_0$ , najdeme další aproximaci řešení  $x^{s+1}$  pomocí vztahu

$$x^{s+1} : \nabla_x P(x^{s+1}; k_s) = 0. \quad (3.4)$$

**Věta 3.3.** *Jestliže jsou množiny  $X^*$  řešení primární úlohy a  $L^*$  řešení duální úlohy omezené, pak*

$$\lim_{k \rightarrow +\infty} f(x^s) = f(x^*), \quad \lim_{k \rightarrow +\infty} \theta(\lambda^s) = \theta(\lambda^*),$$

kde  $x^* \in X^*$ ,  $\lambda^* \in L^*$ .

---

<sup>1</sup>z angl. Path-Following

**Důkaz:** viz [12] str. 123.

Z předchozí věty plyne, že metoda PF je globálně konvergentní metodou.

Pro vyřešení úlohy 3.4 chceme použít Newtonovu metodu s volbou délky kroku. Tedy je zapotřebí, aby funkce  $P(x; k)$  byla silně konvexní, o čemž hovoří následující lemma.

**Lemma 3.2.** *Nechť jsou splněny předpoklady (A) a (C), viz kapitola 2.1, potom existuje  $k_0 > 0$  takové, že pro všechna  $k \geq k_0$  je funkce  $P(x; k)$  silně konvexní v okolí bodu minima  $\hat{x} = \hat{x}(k)$ .*

**Důkaz:** viz [14] str. 17.

**Poznámka 3.3.** Je nutné poznamenat, že pro zaručení silné konvexity funkce  $P(x; k)$ , pro libovolné  $k > 0$ , stačí, aby byla silně konvexní funkce  $f$  nebo aspoň jedna z funkcí  $-c_i(x)$ ,  $i = 1, \dots, r$ .

### 3.3.2 Newtonova metoda pro primárně-duální systém

Budeme postupovat podobně jako v kapitole 2.5. Popíšeme si jeden krok Newtonovy metody pro řešení primárně-duálního systému ve tvaru

$$\begin{aligned} \nabla_x L(\hat{x}; \hat{\lambda}) &= 0, \\ \hat{\lambda} - \Psi'(kc(\hat{x}))e &= 0, \end{aligned} \tag{3.5}$$

jehož řešení je ekvivalentní s řešením soustavy rovnic (3.4).

Předpokládejme, že  $(x, \lambda) \in \mathbb{R}^n \times \mathbb{R}_{++}^r$  je známý odhad primárně-duálního páru  $(x^*, \lambda^*)$ . Provedeme linearizaci soustavy (3.5), tedy

$$\begin{aligned} \hat{x} &= x + \Delta x, \\ \hat{\lambda} &= \lambda + \Delta \lambda, \end{aligned}$$

kde primárně-duální směr  $(\Delta x, \Delta \lambda)$  vypočteme ze soustavy

$$\begin{bmatrix} \nabla_{xx}^2 L(x; \lambda) & -\nabla c(x)^T \\ -k\Psi''(kc(x)) \nabla c(x) & \mathbf{I}_r \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} -\nabla_x L(x; \lambda) \\ 0 \end{bmatrix}. \tag{3.6}$$

Lze říci, že tato soustava je pouze speciálním případem soustavy (2.14), ve které položíme  $\bar{\lambda} := \lambda$ ,  $\lambda := e$ . Díky této skutečnosti platí o soustavě (3.6) analogické úvahy a tvrzení těm uvedeným v kapitole 2.5.

### 3.3.3 Kombinace přístupu NR a PDPF metody

Zvolme libovolný počáteční odhad řešení úlohy (2.2), označme jej  $x^0$ . Dále označme  $\lambda^0 = e$  odhad řešení příslušné duální úlohy. Vhodný postup pro hledání primárně-duálního řešení je takovýto:

1. V první fázi použijeme primárně-duální metodu sledování cesty (PDPF).
2. Jakmile parametr  $k$  bude dostatečně velký, již dále jeho hodnotu zvyšovat nebudeme a přejdeme od PDPF metody k metodě NR. Pro rozhodnutí, kdy k takovéto situaci dojde, se používá hodnotící funkce  $\nu(x, \lambda)$ .
3. V poslední fázi procesu metoda NR automaticky přejde v PDNR metodu.

Výhodou výše popsaného postupu oproti samostatnému použití PDNR nebo PDPF metody je globální konvergence metody – ta je zaručena díky globální konvergenci PDPF metody. Dále pak není zapotřebí neomezeného zvyšování hodnoty parametru  $k$ , a tudíž nedochází k řešení soustav s velmi špatně podmíněnou maticí. Podrobně je celý algoritmus popsán ve [12].

## 3.4 Regularizovaná Newtonova metoda

Alternativou k přístupům popsaných v předchozích dvou kapitolách je použití **regularizované Newtonovy metody** (RNM) namísto čisté Newtonovy metody při řešení soustavy (2.11).

Uvedeme zde pouze základní myšlenky RNM, podrobně je tato metoda vysvětlena v [11]. Nejdříve řekneme, kdy je vhodné RNM používat a k čemu je užitečná ve srovnání s klasickou či tlumenou Newtonovou metodou. Poté začleníme naše úvahy do kontextu NR.

Uvažujme úlohu nepodmíněné minimalizace ve tvaru

$$\min_{x \in \mathbb{R}^n} f(x). \quad (3.7)$$

Jestliže je funkce  $f : \mathcal{D}(f) \rightarrow \mathbb{R}$ ,  $\mathcal{D}(f) \subseteq \mathbb{R}^n$ , dvakrát spojitě diferencovatelná a silně konvexní, pak je vhodné úlohu (3.7) řešit Newtonovou metodou, přičemž globální konvergenci metody zajistíme volbou délky kroku.

RNM, o níž se hovoří níže, lze aplikovat na úlohu (3.7) i v případě, že je účelová funkce pouze konvexní. Navíc je možné její použitelnost rozšířit na funkce, které jsou sice konvexní, avšak nejsou dvakrát spojitě diferencovatelné.

Předpokládejme na začátku, že  $f$  je dvakrát spojitě diferencovatelná, a definujme **regularizaci** funkce  $f$  předpisem

$$F(x, y) = f(y) + \frac{1}{2} \|\nabla f(x)\|_2 \|y - x\|_2^2. \quad (3.8)$$

**Poznámka 3.4.** Je zřejmé, že

- (i)  $F(x, y)|_{y=x} = f(x)$ ,
- (ii)  $\nabla_y F(x, y)|_{y=x} = \nabla f(x)$ ,
- (iii)  $\nabla_{yy}^2 F(x, y)|_{y=x} = \nabla^2 f(x) + \|\nabla f(x)\|_2 \mathbf{I}_n =: H(x)$ .

**Poznámka 3.5.** Uvažujme matici  $H(x)$  z předchozí poznámky. Pokud  $x_0 \in \mathbb{R}^n$  není řešením úlohy (3.7), pak existuje  $[H(x_0)]^{-1}$ , neboť  $\|\nabla f(x_0)\|_2 > 0$ .

Následná RNM je díky poznámce 3.5 dobře definovaná. Při známém odhadu  $x_k \in \mathbb{R}^n$ ,  $k \in \mathbb{N}$ , najdeme následující odhad  $x_{k+1} \in \mathbb{R}^n$  pomocí vztahů

$$\begin{aligned} H(x_k) \Delta x &= -\nabla f(x_k), \\ x_{k+1} &= x_k + \Delta x. \end{aligned} \quad (3.9)$$

Použití a užitečnost RNM nyní prezentujeme na jednoduchém příkladě.

**Příklad 3.2.** Je dána funkce  $f(x) = \ln(e^x + e^{-x})$ . Řešte pomocí klasické Newtonovy metody (NM) a RNM úlohu

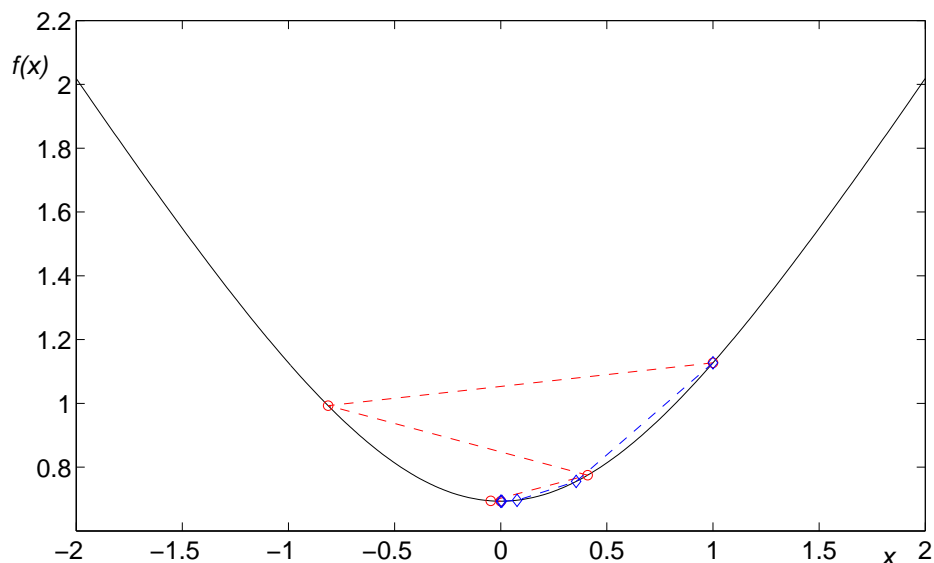
$$\min_{x \in \mathcal{D}(f)} f(x). \quad (3.10)$$

Jako počáteční odhady volte  $x_0 = 1$  a  $\hat{x}_0 = 1, 1$ . Dosažené výsledky porovnejte.



*Řešení:* Jak je zřejmé z předpisu funkce  $f$  a z obrázku 7, účelová funkce má minimum v bodě  $x^* = 0$ .

Podle vztahů (1.14) resp. (3.9) budeme konstruovat posloupnost  $\{x_k\}$ . Při výpočtu použijeme MATLAB a výsledky zapíšeme přehledně do tabulky.



Obrázek 7: Graf funkce  $f(x) = \ln(e^x + e^{-x})$ . Červené body znázorňují postup klasické NM a modré body postup RNM při počátečním odhadu  $x_0 = 1$

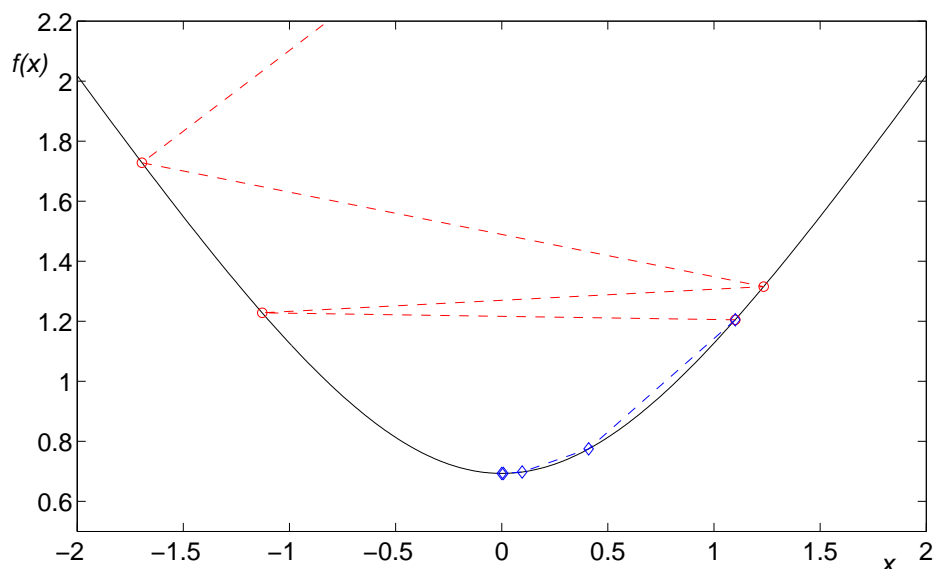
$k$	klasická NM		RNM	
	$x_k$	$\frac{ x_k - x^* }{ x_{k-1} - x^* }$	$x_k$	$\frac{ x_k - x^* }{ x_{k-1} - x^* }$
0	1,0000	—	1,0000	—
1	-0,8134	0,8134	0,3554	0,3554
2	0,4094	0,5033	0,0769	0,2163
3	-0,0473	0,1156	0,0052	0,0680
4	$7,0603 \cdot 10^{-5}$	0,0015	$2,7068 \cdot 10^{-5}$	0,0052

Tabulka 2: Průběh klasické NM a RNM při počátečním odhadu  $x_0 = 1$ .

Při počátečním odhadu  $x_0 = 1$  obě metody konvergují k řešení úlohy (3.10), navíc již po čtyřech krocích jsme v obou případech dosáhli velmi přesného výsledku. Můžeme si všimnout, že charakter konvergence metod je rozdílný. New-

tonova metoda nekonverguje přímo, nýbrž „skáče“ kolem bodu  $x^*$ , zatímco RNM přímo konverguje (viz obr. 7).

Ve druhém případě, při počátečním odhadu  $x_0 = 1,1$ , dochází k tomu, že Newtonova metoda diverguje. Zde se ukáže užitečnost RNM, poněvadž tato metoda s konvergencí problém nemá (viz obr. 8).



Obrázek 8: Graf funkce  $f(x) = \ln(e^x + e^{-x})$ . Červené body znázorňují postup klasické NM a modré body postup RNM při počátečním odhadu  $\hat{x}_0 = 1,1$ .

$k$	klasická NM		RNM	
	$x_k$	$\frac{ x_k - x^* }{ x_{k-1} - x^* }$	$x_k$	$\frac{ x_k - x^* }{ x_{k-1} - x^* }$
0	1,1000	—	1,1000	—
1	-1,1286	1,0260	0,4097	0,3725
2	1,2341	1,0936	0,0960	0,0824
3	-1,6952	1,3736	0,0079	0,0680
4	5,7154	3,3716	$6,1869 \cdot 10^{-5}$	0,0078

Tabulka 3: Průběh klasické NM a RNM při počátečním odhadu  $\hat{x}_0 = 1,1$ . ■

Lze ukázat (viz [11] věta 2), že rychlost konvergence RNM je stejná jako rychlost konvergence NM, tedy kvadratická. Abychom však zaručili globální kon-

vergenci této metody, je opět potřeba volit délku kroku (viz [11] věta 3). V důsledku jsme tedy doposud získali určitou modifikaci tlumené Newtonovy metody pro konvexní a dvakrát spojitě diferencovatelné funkce.

Jestliže účelová funkce v úloze (3.7) je pouze jednou spojitě diferencovatelná, definujeme maticovou funkci

$$A(x) = \begin{cases} \nabla^2 f(x) & , \text{ pokud } \nabla^2 f(x) \text{ existuje,} \\ \mathbf{O}_n & , \text{ jinak,} \end{cases} \quad (3.11)$$

kde  $\mathbf{O}_n \in \mathbb{R}^{n \times n}$  je nulová čtvercová matice.

RNM pak konstruuje posloupnost  $\{x_k\}$  podle rekurentního vztahu

$$x_{k+1} = x_k - \alpha_k (A(x_k) + \|\nabla f(x_k)\|_2 \mathbf{I}_n)^{-1} \nabla f(x_k), \quad (3.12)$$

přičemž  $\alpha_k > 0$  je vhodně zvolená délka  $k$ -tého kroku.

Pokud  $\nabla^2 f(x_k)$ , pro nějaké  $k \in \mathbb{N}_0$ , neexistuje, má vztah (3.12) tvar

$$x_{k+1} = x_k - \alpha_k (\|\nabla f(x_k)\|_2)^{-1} \nabla f(x_k), \quad (3.13)$$

což je metoda největšího spádu, kde délka  $k$ -tého kroku je  $t_k = \alpha_k (\|\nabla f(x_k)\|_2)^{-1}$ .

**Definice 3.4.** Nechť je dána konvexní množina  $\mathcal{D} \subseteq \mathbb{R}^n$ , funkce  $f : \mathcal{D} \rightarrow \mathbb{R}$  a bod  $\hat{x} \in \mathcal{D}$ . Řekneme, že vektor  $g(\hat{x}) \in \mathbb{R}^n$  je **subgradientem** funkce  $f$  v bodě  $\hat{x}$ , jestliže je splněno

$$f(x) - f(\hat{x}) \geq (x - \hat{x})^\top g(\hat{x}), \quad \forall x \in \mathcal{D}.$$

Uvažujme nyní účelovou funkci, jež je konvexní, ale nemusí být vůbec diferencovatelná. Známým poznatkem z konvexní analýzy je existence aspoň jednoho subgradientu konvexní funkce v libovolném bodě jejího definičního oboru.

V případě, že je účelová funkce v bodě  $x_k$ , pro nějaké  $k \in \mathbb{N}_0$ , nediferencovatelná, dosadíme do vztahu (3.13) za gradient účelové funkce její subgradient v bodě  $x_k$ . Toto nám umožňuje používat RNM i v případě, že je účelová funkce nehladká a konvexní. Ve srovnání s tlumenou Newtonovou metodou je tedy RNM použitelná na širší třídu funkcí.

Pro úplnost a přehlednost uvedeme algoritmus RNM.

### Algoritmus 3.4. (RNM)

Předpokládejme, že odhad  $x_k \in \mathbb{R}^n$ ,  $k \in \mathbb{N}_0$ , řešení úlohy nepodmíněné minimalizace dané konvexní funkce  $f$  je znám. Další aproximaci  $x_{k+1}$  nalezneme pomocí postupu:

1. Ze vztahu

$$(A(x_k) + \|v(x_k)\|_2 \mathbf{I}_n) \Delta x = -v(x_k)$$

určíme  $\Delta x$ .

2. Najdeme  $\alpha_k \in (0; 1)$  pomocí metody zpětného vyhledávání.
3. Položíme  $x_{k+1} = x_k + \alpha_k \Delta x$ .

Tento postup opakujeme tak dlouho, dokud není splněno zastavovací kritérium (např.  $-\frac{1}{2} \Delta x^T v(x_k) \leq \varepsilon$ ).

Matice  $A(x_k)$  je definována předpisem (3.11) a vektor  $v(x_k)$  je dán vztahem

$$v(x_k) = \begin{cases} \nabla f(x_k), & \text{pokud } \nabla f(x_k) \text{ existuje,} \\ g(x_k), & \text{jinak,} \end{cases}$$

kde  $g(x_k)$  značí subgradient funkce  $f$  v bodě  $x_k$ .

#### 3.4.1 Použití RNM v teorii NR

Stejně jako je uvedeno v kapitole 1.2, jež hovoří o Newtonově metodě, můžeme uvažovat RNM pro řešení nelineární soustavy rovnic  $F(x) = 0$ , kde  $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ . Nyní nemusíme předpokládat existenci Jacobiho matice zobrazení  $F$ .

Původním úkolem, který jsme si stanovili, bylo vyřešit úlohu konvexního programování (2.2). Jakmile se stane, že účelovou funkci  $f$  nebo některou z funkcí  $c_i$ ,  $i = 1, \dots, r$ , nebude možné derivovat, nemusíme klesat na mysl, neboť použijeme právě RNM namísto klasické Newtonovy metody.

Při řešení soustavy (2.11) tedy nahradíme Newtonovu metodu regularizovanou Newtonovou metodou. Použijeme-li stejného označení jako v kapitole 2.5, nahradíme tak primárně-duální soustavu rovnic (2.14) soustavou

$$(A(\cdot) + \|v(x)\|_2 \mathbf{I}_{n+r}) \begin{bmatrix} \Delta x \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} -v(x) \\ 0 \end{bmatrix}, \quad (3.14)$$

kde

$$A(\cdot) = \begin{cases} N(\cdot), & \text{pokud } N(\cdot) \text{ existuje,} \\ \mathbf{O}_n, & \text{jinak,} \end{cases}$$

$$v(x) = \begin{cases} \nabla_x L(x; \bar{\lambda}), & \text{pokud } \nabla_x L(x; \bar{\lambda}) \text{ existuje,} \\ g_x(x; \bar{\lambda}), & \text{jinak.} \end{cases}$$

Symbolem  $g_x(x; \bar{\lambda})$  je označen subgradient funkce  $L(x; \bar{\lambda})$  v bodě  $x$  pro pevně dané  $\bar{\lambda}$ . Vzhledem k tomu, že funkce  $L(x; \bar{\lambda}) = \mathcal{L}(x; \lambda, k)$  je pro pevnou volbu  $(\lambda, k) \in \mathbb{R}_+^r \times \mathbb{R}_{++}$  konvexní (viz lemma 2.2), víme, že takový subgradient existuje.

Ukážeme aplikaci výše popsaného postupu na jednoduchém příkladě.

**Příklad 3.3.** Jsou dány funkce  $f_1, f_2, f : \mathbb{R} \rightarrow \mathbb{R}$  ve tvaru

$$\begin{aligned} f_1(x) &= \frac{1}{2}x, \\ f_2(x) &= 4x^2 - 4x + 1, \\ f(x) &= \max \{f_1(x), f_2(x)\}. \end{aligned}$$

Pomocí PDNR metody s použitím RNM namísto klasické Newtonovy metody řešte úlohu

$$\begin{cases} \text{minimalizovat } f(x), & x \in \mathbb{R}, \\ \text{za podmínky } & 1 - |x| \geq 0. \end{cases}$$

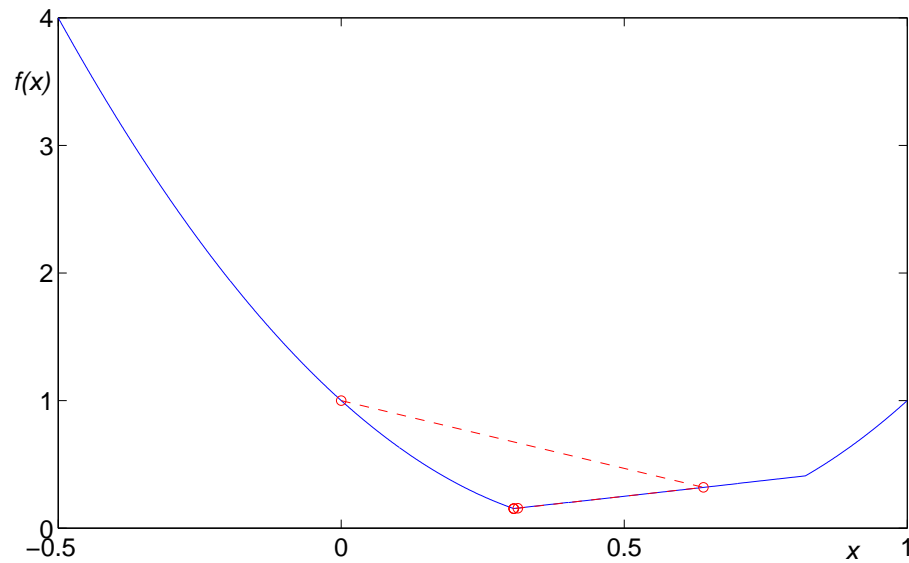
Za počáteční odhad zvolte  $x^0 = 0$ .

*Řešení:* Účelová funkce  $f$  je konvexní, avšak není silně konvexní, na svém definičním oboru. V bodech  $\frac{9 \pm \sqrt{17}}{16}$  je tato funkce nediferencovatelná. Za její subgradient v obou těchto bodech vezmeme například číslo  $\frac{1}{2}$ .

Na obrázku 9 je znázorněn graf funkce  $f$ , z něhož je patrné, že funkce  $f$  má minimum v bodě  $x^* = \frac{9 - \sqrt{17}}{16}$ .

Funkce  $c(x) = 1 - |x|$  nemá derivaci v bodě 0, proto uvažujeme subgradient funkce  $-c(x)$  v tomto bodě a položíme jej roven jedné.

Výpočet provedeme pomocí upraveného M-souboru `pdnr1` ukrytého pod názvem `pdnrReg1`. Poněvadž symbolické funkce použité v `pdnr1` nejsou vhodné pro po částech definované funkce, použijeme namísto nich anonymní funkce. Diskuze o výhodách a nevýhodách použití symbolických či anonymních funkcí je uvedena v kapitole 4.



Obrázek 9: Graf funkce  $f(x) = \max\{f_1(x), f_2(x)\}$ . Červené body znázorňují konvergenci PDNR metody při počátečním odhadu  $x_0 = 0$ .

Hodnotu parametru  $k$  nastavíme na stovku a za počáteční odhad volíme bod  $x_0 = 0$ . Třebaže je funkce  $c(x)$  v tomto bodě nediferencovatelná, nečiní to žádné problémy. M-soubor `pdnrReg1` v MATLABu zavoláme příkazem:

```
[xs, fxs, s] = pdnrReg1(0)
```

Získáme tak výsledek:

```
xs =
    0.3048
fxs =
    0.1524
s =
    7
```

Je potřeba sedmi kroků metody, abychom našli řešení naší úlohy se zvolenou tolerancí přesnosti  $\varepsilon = 10^{-4}$ . Postup konvergence metody je zaznamenán na obrázku 9 červenou čarou. ■

## 4 Příklady

Použitím metody konečných prvků při řešení kontaktních úloh lineární elasticity ve 3D vznikají úlohy nelineárního programování typu

$$\begin{cases} \text{minimalizovat} & \frac{1}{2}x^T Ax - x^T b, & x \in \mathbb{R}^n \\ \text{za podmíněk} & g_i^2 - x_{i+m}^2 - x_{i+2m}^2 \geq 0, & i = 1, \dots, m, \\ & x_i - l_i \geq 0, & i = 1, \dots, m, \end{cases} \quad (4.1)$$

kde  $n = 3m$  je počet neznámých,  $A \in \mathbb{R}^{n \times n}$  je symetrická a pozitivně definitní,  $b \in \mathbb{R}^n$ ,  $g \in \mathbb{R}_+^m$ ,  $l \in \mathbb{R}^m$ . Poněvadž se jedná o úlohu konvexního programování, můžeme k jejímu řešení použít PDNRD metodu.

Bylo by sice možné použít již připravený M-soubor `pdnrd`, avšak lepším přístupem je tento M-soubor modifikovat na míru úloze (4.1). Přestože má M-soubor `pdnrd` tu výhodu, že jej můžeme použít na obecnou úlohu konvexního programování, jsou v něm použity, za účelem výpočtu derivací, symbolické funkce. Práce se symbolickými funkcemi a jejich vyčíslování je v MATLABu časově náročná, a proto pokud předem známe derivace účelové funkce a funkcí podmínek, je vhodné použít anonymní funkce, jež následný běh programu značně urychlí.

Tato kapitola se věnuje řešení dvou úloh typu (4.1) pomocí PDNRD metody. V úvodu je diskutováno nastavení parametrů (viz tabulka 1), jež přímo ovlivňují konvergenci PDNRD metody. Veškeré výpočty jsou prováděny v MATLABu na PC Intel Pentium (1,7 GHz) s 1GB RAM. V následujících tabulkách jsou vždy uvedeny počty iterací, počty řešení primárně-duální soustavy a doba trvání výpočtu. Pokud v některé z tabulek nějaký údaj chybí, znamená to, že výpočet trval příliš dlouho.

### 4.1 O parametrech PDNRD metody

Při použití numerické metody je velice důležité správné nastavení jejích parametrů. Na začátku můžeme mít sice nějaký rozumný odhad, jak parametry nastavit, avšak pouze zkušenosti získané aplikací metody na problémy různého charakteru nám pomohou optimální nastavení parametrů najít.

Na základě testování PDNRD metody na příkladech uvedených v kapitolách 4.2 a 4.3 provedeme nyní úvahy o vhodném nastavení parametrů PDNRD metody.

#### *Faktor $\omega$*

Faktor  $\omega$  ovlivňuje rychlost zvyšování parametru rescalingu. Jako první nápad se nabízí položit jej roven dvěma či desíti. Při řešení příkladů lze zjistit, že pro hodnoty  $\omega \in \langle 5; 20 \rangle$  dostáváme srovnatelné výsledky. Budeme tedy volit  $\omega = 10$ . Avšak ani v případě jiné volby  $\omega > 1$  nedojde k výrazným změnám v průběhu výpočtu. Nanejvýš se může stát, že bude potřeba několika kroků metody navíc.

#### *Parametr $\sigma$*

Není ani tak důležité, jak zvolíme parametr  $\sigma$ , ale jak se rozhodneme zvolit poměr mezi  $\sigma$  a počáteční volbou parametru rescalingu  $k_{init}$ . Vždyť právě podíl  $\frac{\sigma}{k}$  rozhoduje o počtu vnitřních kroků metody (tj. o počtu kroků Newtonovy tlumené metody). Je zřejmé, že pro  $\sigma \gg k$  bude vnitřních kroků příliš málo. Na druhou stranu pro  $\sigma \ll k$  jich bude zase příliš mnoho. V souladu s tímto se volba  $\sigma = \frac{1}{2}k_{init}$  či  $\sigma = \frac{1}{5}k_{init}$  ukazuje jako vhodná.

#### *Parametr $\theta$*

Parametr  $\theta$  ovlivňuje, zda bude spuštěn vnitřní řešič či nikoliv. Pokud položíme  $\theta = 0$ , říkáme tím, že vnitřní řešič spustíme vždy, když nedojde k 1,5-superlineárnímu poklesu hodnotící funkce. Při volbě  $\theta = 0,5$  říkáme, že nám stačí, aby v každém kroku došlo k lineárnímu poklesu hodnotící funkce. Je tudíž rozumné volit  $\theta \in \langle 0; 0,5 \rangle$ . Ukazuje se (viz tabulky 6 a 9), že na volbě tohoto parametru úspěšnost metody příliš nezávisí.

V podmínce týkající se parametru  $\theta$  se počítá  $\min \{H^{3/2-\theta}, 1 - \theta\}$ . Rovněž by stačilo namísto tohoto minima uvažovat jednoduše hodnotu  $H^{3/2-\theta}$ , avšak jak víme, klasická Newtonova metoda je efektivní pouze v okolí bodu řešení, a tak pokud jsme ještě „daleko“ od řešení, je vhodnější použít tlumenou Newtonovu metodu. Výraz „daleko“ je myšlen tak, že hodnotící funkce nabývá hodnoty větší než  $1 - \theta$ .



### *Faktor $q$*

Společně s podílem  $\frac{\sigma}{k}$  ovlivňuje počet vnitřních kroků i faktor  $q$ . Tento faktor má navíc vliv na to, jak často se bude zvětšovat parametr rescalingu. Vzhledem k charakteru podmínky v bodě (8°) algoritmu 3.2 je potřeba, aby  $q \in (0; 1)$ . Navíc musíme faktor  $q$  volit tak, aby metoda nepoužívala příliš mnoho kroků. V některých případech (viz tabulka 6) by totiž výpočet trval dlouho. Na základě řešení příkladů pro různé hodnoty parametru  $q$  (viz tabulky 6 a 9) doporučuje autor nenastavovat jej příliš malý.

Hodnoty parametru  $q$  z intervalu  $\langle 0, 5; 1 \rangle$  dávají srovnatelné výsledky, proto se vhodnou ukazuje volba  $q = 0,5$ .

### *Parametr $\eta$*

Parametr zpětného vyhledávání  $\eta$  se zpravidla volí v rozmezí 0,01 až 0,3 (viz [4] str. 466). Jak je vidět z tabulek 4, 5 a 6, volba tohoto parametru nemá při řešení úlohy se strunou žádný vliv na běh PDNRD metody. Řešíme-li však problém s ocelovým kvádrem, parametr  $\eta$  již má určitý vliv na počet iterací PDNRD metody a také na počet řešení primárně-duální soustavy (viz tabulky 7, 8 a 9). Lepší se v tomto případě jeví nastavení  $\eta = 0,01$ .

### *Volba $k_{init}$*

Na základě výsledků uvedených v tabulkách 4 a 5 by se mohlo zdát, že s rostoucím počtem proměnných v úloze je potřeba zvyšovat i počáteční hodnotu parametru rescalingu. Avšak při řešení problému s ocelovým kvádrem se toto nepotvrdilo, poněvadž pro  $n = 648$  je vhodná volba  $k_{init} = 2 \cdot 10^2$  (viz tabulka 8).

Každopádně lze říci, že pro  $k_{init} \in \langle 10^3; 10^5 \rangle$  dosáhneme řešení zpravidla po přijatelném počtu kroků. Pro menší nebo výrazně větší hodnoty parametru rescalingu již výpočet trvá déle (řádově desítky kroků).

### *Volba počátečního odhadu*

Průběh výpočtu závisí i na volbě počátečního odhadu. Ačkoliv již víme, že PDNRD metoda je globálně konvergentní metodou, vhodná volba počátečního odhadu může počet kroků potřebných k dosažení požadované přesnosti značně

snížit. Při řešení takových úloh, jako jsou například průhyby nosníku nebo kontaktní úlohy, se nabízí volit za počáteční odhad výchozí stav systému, tj. bez působení vnějších sil. Je to z toho důvodu, že změny tvarů těles v systému jsou zpravidla velmi malé. Budeme tedy volit  $x^0 = (0, 0, \dots, 0)$ .

## 4.2 Problém se strunou

Uvažujme úlohu ve tvaru

$$\min_{u \in \mathcal{K}} \mathcal{J}(u), \quad (4.2)$$

kde

$$\mathcal{J}(u) = \frac{1}{2} \int_0^1 \|u'(t)\|^2 dt - \int_0^1 u(t)^T f(t) dt,$$

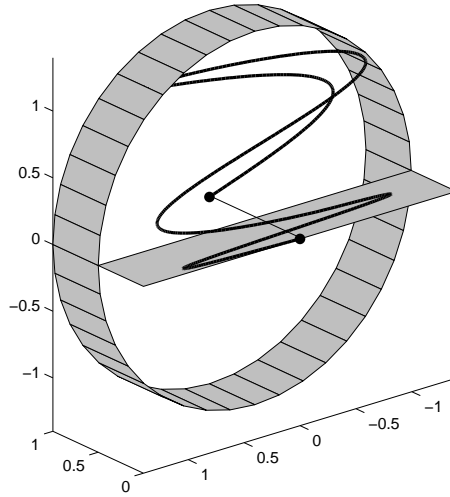
$$\mathcal{K} = \left\{ u \in (H_0^1(0; 1))^2 : u_2(t) \geq 0, \forall t \in (0; 0,5), \|u(t)\| \leq 1,4, \forall t \in (0,5; 1) \right\},$$

$$f(t) = (36\pi^2 \sin 6\pi t, -4\pi^2 \sin 2\pi t)^T.$$

Minimalizační problém (4.2) popisuje zatíženou strunu, která je pevně uchylena v koncových bodech, částečně umístěna nad rovinou a částečně uvnitř válcové plochy (viz obrázek 10). Funkce  $u(t)$  označuje vychýlení struny a funkce  $f(t)$  její zatížení.

Aproximací úlohy (4.2) metodou konečných prvků, pomocí spojitých po částech lineárních funkcí na ekvidistantní síti s  $n = 4m$  stupni volnosti, získáme úlohu konvexního programování velice podobnou úloze (4.1). Jediným rozdílem je, že kromě podmíněných proměnných  $x_i$ ,  $i = 1, \dots, 3m$ , jsou v této úloze navíc nepodmíněné proměnné  $x_i$ ,  $i = 3m + 1, \dots, 4m$ . Tyto nepodmíněné proměnné však z teoretického a následně z výpočetního hlediska nečiní žádné problémy.

Účelová funkce  $\frac{1}{2}x^T Ax - x^T b$ ,  $x \in \mathbb{R}^n$ , odpovídá konvexnímu kvadratickému funkcionalu  $\mathcal{J}(u)$ , lineární podmínky v úloze (4.1) pak podmínce  $u_2(t) \geq 0$ ,  $\forall t \in (0; 0,5)$  z definice přípustné množiny  $\mathcal{K}$  a kvadratické podmínky podmínce  $\|u(t)\| \leq 1,4$ ,  $\forall t \in (0,5; 1)$ .



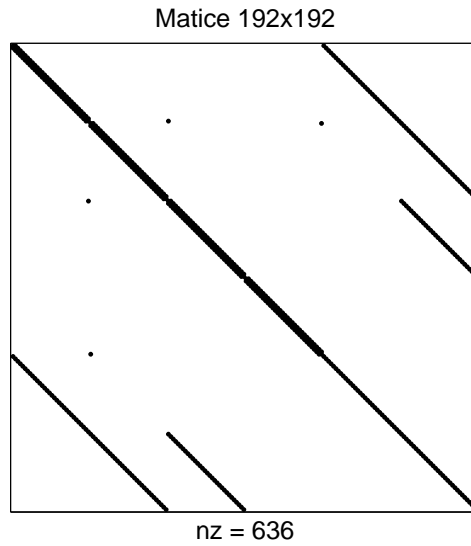
Obrázek 10: Deformace struny.

*Řešení:* Úlohu vyřešíme pro různé počty proměnných, abychom otestovali rychlost PDNRD metody. Ze všeho nejdříve je však nutné určit, zda matice primárně-duální soustavy  $N_k(\cdot)$  je řídkou maticí, abychom se mohli rozhodnout, který z přístupů popsaných v kapitole 2.5 použijeme.

Za řídkou maticí se zpravidla považuje taková matice, která má nejvýše 10% nenulových prvků. Na obrázku 11 je vidět struktura matice  $N_k(\cdot)$  pro  $n = 128$ . Tato matice má rozměry  $192 \times 192$ , z čehož pouze 636 prvků je nenulových. Matici  $N_k(\cdot)$  tudíž můžeme označit za řídkou, protože má jen 1,7% nenulových prvků. Analogicky i pro jiné volby počtu proměnných  $n$  je matice  $N_k(\cdot)$  řídká. Budeme tedy řešit primárně-duální soustavu jako celek s použitím některé z technik pro řešení soustav s řídkou maticí.

Zvolíme toleranci přesnosti  $\varepsilon = 10^{-6}$  a parametry PDNRD metody podle kapitoly 4.1. Úlohu vyřešíme pomocí M-souboru `pdnrd_spec.m`, jenž se volá příkazem

```
[x_nr, f_nr, s] = pdnrd_spec(A, b, l, g, zeros(length(b), 1));
```



Obrázek 11: Struktura matice primárně-duální soustavy při řešení problému se strunou. Černou barvou jsou znázorněny nenulové prvky.

Označme  $n = 4m$  počet proměnných a  $r = 2m$  počet podmínek. Postupně řešíme zadaný problém pro  $n = 2^6, 2^7, \dots, 2^{12}$  a různé volby  $k_{init}$ . Vše zapíšeme do tabulek 4 a 5.

$\eta = 0,01$		$k_{init}$			
$n$	$r$	$2 \cdot 10^2$	$2 \cdot 10^3$	$2 \cdot 10^4$	$2 \cdot 10^5$
64	32	8/13/0,359	6/6/0,063	7/12/0,109	6/14/0,203
128	64	9/21/0,547	7/10/0,328	7/12/0,234	6/12/0,219
256	128	–	6/19/1,515	–	4/10/0,532
512	256	–	–	5/10/3,156	4/12/3,094
1024	512	–	–	5/13/19,890	3/6/9,578
2048	1024	–	–	–	4/7/62,375
4096	2048	–	–	–	4/9/553,481

Tabulka 4: PDNRD metoda s parametry  $\omega = 10$ ,  $\sigma = \frac{1}{2}k_{init}$ ,  $\theta = 0,4$ ,  $q = 0,5$ ,  $\eta = 0,01$ ,  $\varepsilon = 10^{-6}$ . Problém se strunou.

$\eta = 0,3$		$k_{init}$			
$n$	$r$	$2 \cdot 10^2$	$2 \cdot 10^3$	$2 \cdot 10^4$	$2 \cdot 10^5$
64	32	8/13/0,375	6/6/0,078	7/12/0,156	6/14/0,187
128	64	9/20/0,578	7/10/0,218	7/12/0,219	6/12/0,282
256	128	–	6/19/1,297	–	4/10/0,437
512	256	–	–	5/10/3,047	4/12/3,141
1024	512	–	–	5/13/20,469	3/6/9,063
2048	1024	–	–	–	4/7/63,484
4096	2048	–	–	–	4/9/551,516

Tabulka 5: PDNRD metoda s parametry  $\omega = 10$ ,  $\sigma = \frac{1}{2}k_{init}$ ,  $\theta = 0,4$ ,  $q = 0,5$ ,  $\eta = 0,3$ ,  $\varepsilon = 10^{-6}$ . Problém se strunou.

Při pohledu do tabulek 4 a 5 lze učinit závěr, že parametr  $\eta$  nemá v tomto případě na běh PDNRD metody žádný vliv. V tabulce 4 vidíme, že pro větší počet proměnných je vhodné u této úlohy volit větší počáteční hodnotu parametru rescalingu. Druhým, neméně zajímavým, poznatkem je nerostoucí – spíše klesající – počet iterací i počet řešení primárně-duální soustavy s rostoucím počtem proměnných.

Zvolme nyní  $n = 1024$ ,  $k_{init} = 2 \cdot 10^5$  a sledujme průběh výpočtu při různých volbách parametrů  $\eta$ ,  $\theta$ ,  $q$ . Zjištěné údaje zapíšeme do tabulky 5.

$q$	$\eta = 0,01$		$\eta = 0,3$	
	$\theta = 0,1$	$\theta = 0,4$	$\theta = 0,1$	$\theta = 0,4$
0,1	–	–	–	–
0,5	3/6/9,094	3/6/9,078	3/6/9,109	3/6/9,078
0,9	3/6/9,094	3/6/9,079	3/6/9,125	3/6/9,109

Tabulka 6: PDNRD metoda s parametry  $\omega = 10$ ,  $\sigma = \frac{1}{2}k_{init}$ ,  $k_{init} = 2 \cdot 10^5$ ,  $\varepsilon = 10^{-6}$ . Problém se strunou ( $n = 1024$ ).

Jak je vidět z tabulky 6, parametry  $\eta$ ,  $\theta$  nemají na průběh výpočtu žádný vliv. Oproti tomu správná volba parametru  $q$  nám zajistí dokončení výpočtu v rozumném čase. ■

### 4.3 Problém s ocelovým kvádrem

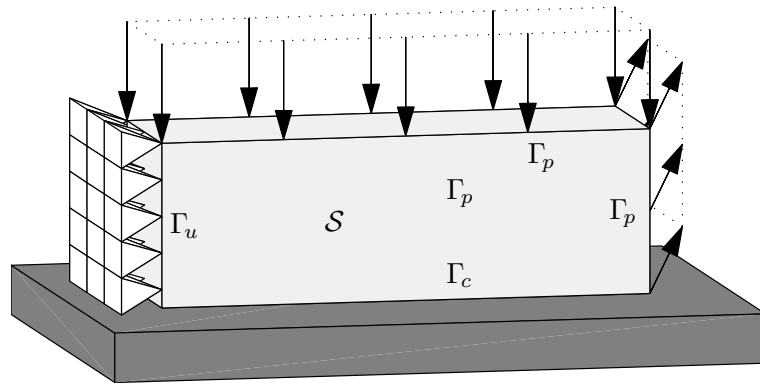
Na úlohu typu (4.1) vede rovněž problém s ocelovým kvádrem ležícím na pevné podložce. Předpokládejme, že kvádr zaujímá prostor  $\mathcal{S} = (0; 3) \times (0; 1) \times (0; 1)$ . Jeho hranice  $\partial\mathcal{S}$  je rozdělena na tři části

$$\Gamma_u = \{0\} \times (0; 1) \times (0; 1),$$

$$\Gamma_c = (0; 3) \times (0; 1) \times \{0\},$$

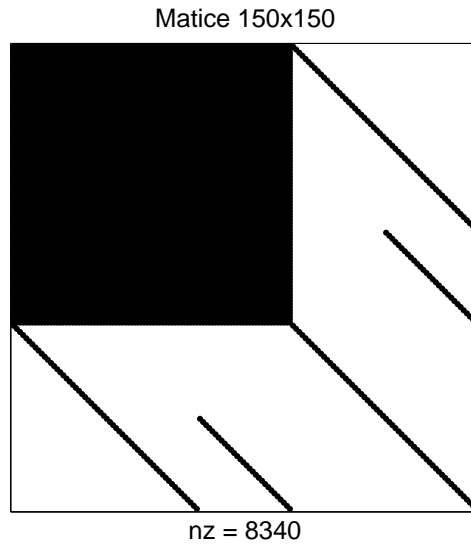
$$\Gamma_p = \partial\mathcal{S} \setminus (\bar{\Gamma}_u \cup \bar{\Gamma}_c),$$

na kterých jsou zadány různé okrajové podmínky (viz obrázek 12). Podrobně je zadání úlohy a následná aproximace metodou konečných prvků popsána v [7].



Obrázek 12: Ocelový kvádr.

*Řešení:* Postupujeme podobně jako při řešení problému se strunou. Nejprve určíme, zda je matice  $N_k(\cdot)$  opět řídká. Na obrázku 13 je znázorněna struktura matice  $N_k(\cdot)$  pro  $n = 90$ . Tato matice má rozměry  $150 \times 150$  a 8340 nenulových prvků. To znamená, že v tomto případě o řídké matici hovořit nemůžeme, neboť má 37% nenulových prvků. Namísto přímého řešení primárně-duální soustavy tedy nejdříve eliminujeme neznámou  $\Delta\lambda$  a vyřešíme soustavu se symetrickou pozitivně definitní maticí  $M_k(\cdot)$  pro neznámou  $\Delta x$ . Newtonův krok pro duální proměnnou následně určíme ze vztahu (2.16).



Obrázek 13: Struktura matice primárně-duální soustavy při řešení problému s ocelovým kvádrem. Černou barvou jsou znázorněny nenulové prvky.

Označíme  $n = 3m$  počet proměnných a  $r = 2m$  počet podmínek. Zvolíme toleranci přesnosti  $\varepsilon = 10^{-6}$  a parametry PDNRD metody podle doporučení v kapitole 4.1. Parametr  $\eta$  položíme roven 0,01 (resp. 0,3).

Problém vyřešíme pro různé hodnoty  $n$  a různé volby parametru  $k_{init}$  pomocí M-souboru `pdnrd_spec2.m`, který se od `pdnrd_spec.m` liší pouze stylem řešení primárně-duální soustavy. Do tabulek 7 a 8 zapíšeme zjištěné údaje.

$\eta = 0,01$		$k_{init}$			
$n$	$r$	$2 \cdot 10^2$	$2 \cdot 10^3$	$2 \cdot 10^4$	$2 \cdot 10^5$
54	36	–	5/85/1,578	9/959/9,391	–
90	60	–	6/47/1,031	6/329/5,047	–
180	120	–	5/31/2,328	8/641/27,718	–
324	216	–	7/58/15,469	5/105/21,234	–
648	432	8/32/50,000	6/30/53,563	6/74/113,454	8/501/800,297

Tabulka 7: PDNRD metoda s parametry  $\omega = 10$ ,  $\sigma = \frac{1}{2}k_{init}$ ,  $\theta = 0,4$ ,  $q = 0,5$ ,  $\eta = 0,01$ ,  $\varepsilon = 10^{-6}$ . Problém s ocelovým kvádrem.

$\eta = 0,3$		$k_{init}$			
$n$	$r$	$2 \cdot 10^2$	$2 \cdot 10^3$	$2 \cdot 10^4$	$2 \cdot 10^5$
54	36	7/33/0,531	5/82/1,328	11/1167/9,578	–
90	60	–	6/39/0,969	6/340/5,640	–
180	120	–	7/48/3,328	10/717/39,594	–
324	216	–	–	5/116/25,922	–
648	432	8/33/59,484	–	6/106/145,313	9/609/891,375

Tabulka 8: PDNRD metoda s parametry  $\omega = 10$ ,  $\sigma = \frac{1}{2}k_{init}$ ,  $\theta = 0,4$ ,  $q = 0,5$ ,  $\eta = 0,3$ ,  $\varepsilon = 10^{-6}$ . Problém s ocelovým kvádrem.

Srovnáme-li údaje uvedené v tabulkách 7 a 8, zjistíme, že nastavení parametru  $\eta = 0,01$  je lepší než volba  $\eta = 0,3$  v tom smyslu, že k výpočtu je potřeba nižšího počtu řešení primárně-duální soustavy. Z tabulky 7 můžeme rovněž odvodit, že pro  $\eta = 0,01$  je výhodné zvolit  $k_{init}$  řádově v rozmezí  $10^2 - 10^3$ .

Z posledního řádku tabulek 7 a 8 plyne, že v tomto případě nelze doporučit s rostoucím  $n$  zvyšovat i parametr  $k_{init}$ . V platnosti však zůstává, že s rostoucím počtem proměnných nedochází ke zvyšování počtu řešení primárně-duální soustavy.

Položme nyní  $n = 324$ ,  $k_{init} = 2 \cdot 10^4$ . Pro různé volby parametrů  $\eta$ ,  $\theta$ ,  $q$  sledujme průběh výpočtu.

$q$	$\eta = 0,01$		$\eta = 0,3$	
	$\theta = 0,1$	$\theta = 0,4$	$\theta = 0,1$	$\theta = 0,4$
0,1	5/105/23,328	5/105/23,312	5/116/25,812	5/116/25,735
0,5	5/105/24,141	5/105/23,984	5/116/25,610	5/116/25,984
0,9	5/105/23,203	5/105/23,437	5/116/25,734	5/116/25,781

Tabulka 9: PDNRD metoda s parametry  $\omega = 10$ ,  $\sigma = \frac{1}{2}k_{init}$ ,  $k_{init} = 2 \cdot 10^4$ ,  $\varepsilon = 10^{-6}$ . Problém s ocelovým kvádrem ( $n = 324$ ).

Z tabulky 9 je vidět, že zatímco testované hodnoty parametrů  $\theta$ ,  $q$  průběh výpočtu nijak neovlivnily, nastavení  $\eta = 0,01$  způsobilo oproti volbě  $\eta = 0,3$  mírné snížení počtu řešení primárně-duální soustavy.

Podívejme se blíže na průběh PDNRD metody pro  $n = 324$ . Zvolme  $\varepsilon = 10^{-12}$ . Při výpočtu nám MATLAB v každém kroku vypíše počet vnitřních kroků metody (`No_inner_iter`), hodnotu hodnotící funkce (`Error`) a parametr rescalingu (`k`).



Variables = 324

Constraints = 216

```
-----
```

Iter	No_inner_iter	Error	k
0	0	1.04e+001	2000
1	0	5.26e-002	2000
2	0	5.67e-003	2000
3	39	4.25e-004	20000
4	9	4.56e-005	20000
5	3	2.96e-006	20000
6	2	4.32e-009	20000
7	1	2.34e-010	6.53e+004
8	0	2.70e-012	6.09e+005
9	0	2.31e-013	2.08e+006

```
-----
```

PD systems solved: 63

Time: 18.61 s

Metoda PDNRD je charakteristická tím, že od určitého kroku – zde konkrétně od osmého – dochází ke žhavému startu metody. Přestože nelze obecně říci, kdy k němu dojde, můžeme očekávat, že náročnost výpočtu s přesností  $10^{-12}$  nebude výrazně vyšší než s přesností například  $10^{-6}$ . To potvrzuje i srovnání počtu řešení primárně-duální soustavy v tomto případě s údajem v tabulce 7.

■

Při použití PDNRD metody je nutné počítat s tím, že tato metoda nepracuje pouze s vnitřními, ale i s vnějšími body. To při nastavení tolerance přesnosti například na hodnotu  $10^{-2}$  může mít za následek, že řešení, jež by správně mělo ležet na hranici přípustné množiny, leží malý kousek za ní (tj. mimo přípustnou množinu). Z tohoto důvodu jsme při testování PDNRD metody volili toleranci přesnosti  $\varepsilon = 10^{-6}$ .

Závěrem je důležité poznamenat, že při zvyšování počtu neznámých v zadané úloze, nedochází k nárůstu počtu řešení primárně-duální soustavy (viz tabulky 4 a 6). Tento fakt je velice důležitý, neboť právě řešení primárně-duální soustavy je časově nejnáročnější částí algoritmu 3.2, a tedy počet řešení primárně-duální soustavy určuje celkovou komplexitu PDNRD metody.

## Závěr

V teoretické části této práce jsme se nejprve seznámili se základy teorie NR. Uvažovali jsme pevně daný parametr rescalingu  $k > 0$  a věnovali se popisu a analýze metod NR a PDNR. Poté jsme zaměřili svou pozornost na vylepšení PDNR metody. Pomocí dynamické změny parametru rescalingu a volby délky kroku jsme dosáhli globální 1,5-Q-superlineární konvergence PDNRD metody.

V části praktické jsme testovali PDNRD metodu a hledali vhodné nastavení jejích parametrů. Ukázali jsme si například, že žhavý start předpovězený větou 2.5 se v teorii NR skutečně vyskytuje. Rovněž jsme pochopili význam jednotlivých parametrů metody.

Teorie NR dosud má mnoho otevřených otázek, které se nabízejí k dalšímu výzkumu v této oblasti. Jednou z nich je například otázka asymptoticky kvadratické rychlosti konvergence PDNRD metody. Existuje totiž důvod si myslet, že metoda PDNRD má potenciál této rychlosti dosáhnout (viz [13] str. 258).

Dalším problémem je zobecnění PDNRD metody pro nekonvexní úlohy. Zda je to vůbec možné, a jakým způsobem to udělat. Také bychom se mohli vydat opačným směrem, tj. hledání optimálního nastavení parametrů PDNRD metody pro různé typy úloh konvexního programování.

Jistě by bylo zajímavé podívat se na souvislost teorie NR a sebeomezujících funkcí, poněvadž při nelineárním rescalingu úlohy v této práci používáme modifikovanou logaritmickou transformaci  $\psi_{q_2}$  (viz poznámka 2.5).

Rovněž je důležité zaměřit se na nalezení metody, jež by efektivně vyřešila primárně-duální soustavu, neboť řešení této soustavy je časově nejnáročnější částí PDNRD metody.

Těmito a dalšími otázkami z teorie optimalizace bych se rád zabýval i nadále. Zejména mě zajímá použití sebeomezujících funkcí v teorii NR a dále pak oblast tvarové optimalizace.

## Literatura

- [1] Alber M., Reemtsen R.: *Intensity modulated radiotherapy treatment planning by use of a barrier-penalty multiplier method*, Optimization Methods and Software, Vol. 22, No. 3, June 2007, str. 391–411
- [2] Andrášik R.: *Newtonova metoda pro úlohu nepodmíněné optimalizace*, Bakalářská práce, Univerzita Palackého v Olomouci, 2010
- [3] Bazaraa M.S., Sherali H.D., Shetty C.M.: *Nonlinear Programming: Theory and Algorithms*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2006
- [4] Boyd S., Vandenberghe L.: *Convex Optimization*, Cambridge University Press, Cambridge, 2004
- [5] Dantzig G.B.: *Linear Programming and Extensions*, Princeton University Press, Princeton, New Jersey, 1998
- [6] Griva I., Nash S.G., Sofer A.: *Linear and Nonlinear Optimization*, Second edition, SIAM, Philadelphia, 2009
- [7] Kučera R., Machalová J., Netuka H., Ženčák P.: *An interior-point algorithm for the minimization arising from 3D contact problems with friction*, Optimization methods and Software, Vol. 00, No. 00, pp. 1–23, 2011
- [8] Nocedal J., Wright S.J.: *Numerical optimization*, Second edition, Springer, 2006
- [9] Polyak R.: *Log-Sigmoid Multipliers Method in Constrained Optimization*, Annals of Operations Research, Vol. 101, pp. 427–460, 2001
- [10] Polyak R.: *Nonlinear rescaling vs. Smoothing Technique in Convex Optimization*, Mathematical Programming, Vol. 92A, pp. 197–235, 2002
- [11] Polyak R.: *Regularized Newton Method for Unconstrained Convex Optimization*, Mathematical Programming, Vol. 120, pp. 125–145, 2009
- [12] Polyak R., Griva I.: *Primal-Dual Nonlinear Rescaling Method for Convex Optimization*, JOTA, Vol. 122, No. 1, pp. 111–156, 2004
- [13] Polyak R., Griva I.: *Primal-Dual Nonlinear Rescaling Method with dynamic scaling parameter update*, Mathematical Programming, Vol. 106A, pp. 237–259, 2006
- [14] Polyak R., Griva I.: *Nonlinear Rescaling in Constrained Optimization: Primal-Dual Approach*, Technical Report SEOR-02-05, Fairfax, Virginia, 2002

- [15] Polyak R., Teboulle M.: *Nonlinear Rescaling and Proximal-Like Methods in Convex Optimization*, Mathematical programming, Vol. 76, pp. 265–284, 1997
- [16] Wright S.J.: *Primal-Dual Interior Point Methods*, SIAM, Philadelphia, 1997