Optimalizace

Petr Mejzlík

Fakulta informatiky MU

FACULTAS ARTIS INFORMATICAE • UNIVERSITAS MASARYKIANA •

Brno 18. prosince 1996

Obsah

1 Úvod

Tato přednáška se týká *minimalizace reálných funkcí*, tj. úlohy

$$\min_{x \in M} f(\mathbf{x}), \text{ kde } f : \mathbb{R}^n \to \mathbb{R} \text{ a } M \subseteq \mathbb{R}^n.$$

Funkce f se pak označuje jako *účelová funkce.*

Není nutné zabývat se samostatně také *maximalizací*, neboť ji lze převést na minimalizaci pomocí vztahu

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

1.1 Motivační příklady

Optimalizační úlohy se vyskytují všude tam, kde je možné vybírat si z více možných rozhodnutí a přitom kvalitu jednotlivých rozhodnutí ohodnotit nějakým reálným číslem. Typickými jsou zde úlohy řízení výroby a dopravy nebo návrh technických zařízení nebo technologií. Optimalizační úlohy hrají také hlavní roli například při prokládání naměřených hodnot (tedy u regresní analýzy), nebo při modelování molekul v chemii.

Aproximace dat: nelineární model bez omezení

Najděte nejlepší aproximaci pomocí součtu čtverců pro funkci

$$\phi(t, \mathbf{x}) = (1 - x_1 t / x_2)^{(1/(x_1 c)) - 1}$$

kde $c = 96.05$ a aproximované body jsou dány tabulkou

t_i	2000	5000	10000	20000	30000	50000
d_i	0.9427	0.8616	0.7384	0.5362	0.3739	0.3096

Plánování výroby: lineární model

Výrobce používá materiál (m) a práci (l) k výrobě nejvýše čtyř výrobků (a až d). Požadavky na jednotlivé výrobky a zisk z jejich prodeje jsou dány následovně:

k výrobě 1 kusu výrobku a je třeba $4m + 2l$ zdrojů a zisk je 50Kč/kus
k výrobě 1 kusu výrobku b je třeba $m + 5l$ zdrojů a zisk je 80Kč/kus
k výrobě 1 kusu výrobku c je třeba $2m + l$ zdrojů a zisk je 30Kč/kus
k výrobě 1 kusu výrobku d je třeba $2m + 3l$ zdrojů a zisk je 40Kč/kus

Během jednoho dne je k dispozici nejvýše 30 jednotek materiálu a 50 jednotek (např. hodin) práce. V jakém počtu mají být produkovány jednotlivé výrobky tak, aby bylo dosaženo maximálního zisku?

Konstrukční problém: nelineární model s omezeními

Navrhněte válcovitou plechovku o objemu 1 dm^3, na kterou se spotřebuje co nejméně kovu.

Aplikace v chemii: nelineární model s více lokálními minimy

Je dán strukturní vzorec molekuly (tj. obrázek, v němž jsou vyznačeny vazby mezi atomy). Určete třírozměrnou strukturu (konformaci) této molekuly za jistých standarních fyzikálních podmínek. Nejjednodušším případem je určení optimální konformace shluku N identických atomů bez elektrostatického náboje ve vakuu. Pak jde o minimalizaci funkce

$$\sum_{ij}\left[\frac{A}{r_{ij}^{12}}-\frac{B}{r_{ij}^{6}}\right],$$

kde A a B jsou jisté kladné reálné konstanty a r_{ij} je vzdálenost i-tého a j-tého atomu.

1.2 Matematické základy

Vektory budeme označovat malými tučnými písmeny (například $\mathbf{a}$) a bude se přitom zpravidla jednat o sloupcové vektory. *Matice* budeme označovat velkými tučnými písmeny (například $\mathbf{B}$). Horní index $^{\mathrm{T}}$ bude znamenat transpozici vektoru nebo matice. Tedy

$$\mathbf{a}=(a_1,a_2,\ldots,a_n)^{\mathrm{T}},\qquad \mathbf{B}=\begin{bmatrix} B_{11} & B_{12} & \ldots & B_{1n}\\ B_{21} & B_{22} & \ldots & B_{2n}\\ \vdots & \vdots & & \vdots\\ B_{m1} & B_{m2} & \ldots & B_{mn}\end{bmatrix}$$

Skalární součin vektorů $\mathbf{a}$ a $\mathbf{z}$ pak zapíšeme jako

$$\mathbf{a}^{\mathrm{T}}\mathbf{z}=\mathbf{z}^{\mathrm{T}}\mathbf{a}=\sum_i a_i z_i.$$

Často se budeme zabývat posloupnostmi bodů z $\mathbb{R}^n$. Jednotlivé body v posloupnosti pak budeme rozlišovat horním indexem v závorce. Budeme je tedy psát například jako $\mathbf{x}^{(1)}$, $x^{(2)}$, $\ldots$ nebo $\{\mathbf{x}^{(k)}\}$. Speciální význam bude mít hvězdička (například v $\mathbf{x}^*$). Bude označovat bod, který je řešením vyšetřovaného problému.

Přímka v $\mathbb{R}^n$ je definována jako množina bodů

$$\mathbf{x}=\mathbf{x}(\alpha)=\mathbf{x}'+\alpha\mathbf{s} \tag{1.1}$$

pro všechna $\alpha\in\mathbb{R}$, kde $\mathbf{x}'$ je jistý bod ležící na přímce a $\mathbf{s}\in\mathbb{R}^n$ je *směr* přímky. Kvůli jednoznačnosti je někdy vhodné směr *normalizovat*, takže například při euklidovské normě je $\mathbf{s}^{\mathrm{T}}\mathbf{s}=\sum_i s_i^2=1$.

U funkcí, kterými se budeme zabývat, budeme vesměs předpokládat, že mají spojité derivace až do druhého řádu. Vektor prvních parciálních derivací v bodě $\mathbf{x}$ budeme pak označovat jako

$$\nabla f(\mathbf{x})=\left(\frac{\partial f}{\partial x_1},\frac{\partial f}{\partial x_2},\ldots,\frac{\partial f}{\partial x_n}\right)\bigg|_{\mathbf{x}}. \tag{1.2}$$

Matici druhých parciálních derivací v bodě $\mathbf{x}$, neboli *Hessovu matici* (také se používá termín *hessián*) budeme označovat symbolem $\nabla^2 f(\mathbf{x})$. i,j-tý prvek této matice je roven $\partial^2 f/(\partial x_i\partial x_j)$. Lze snadno ukázat, že $\nabla^2 f(\mathbf{x})$ je symetrická matice. Je dobré si uvědomit, že operátor ∇ je zobrazením, které každé diferencovatelné reálné funkci f přiřazuje jistou jinou funkci ∇f. Tedy výrazy $\nabla f(\mathbf{x})$ a $\nabla^2 f(\mathbf{x})$ bychom mohli psát i jako $(\nabla f)(\mathbf{x})$ a $(\nabla^2 f)(\mathbf{x})$.

Vyjádření gradientu a Hessovy matice můžeme ilustrovat na Rosenbrockově funkci

$$f(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2, \tag{1.3}$$

která se často používá pro testování kvality optimalizačních metod. V tomto případě je

$$\nabla f(\mathbf{x}) = \left(-400x_1(x_2 - x_1^2) - 2(1 - x_1),\ 200(x_2 - x_1^2)\right)^{\mathrm{T}}$$

$$\nabla^2 f(\mathbf{x}) = \begin{bmatrix} 1200x_1^2 - 400x_2 + 2 & -400x_1 \\ -400x_1 & 200 \end{bmatrix}.$$

Pomocí gradientu ∇ se dají elegantně vyjádřit *směrové derivace*. Derivace funkce f ve směru $\mathbf{s}$ v bodě $\mathbf{x}$ je definována jako

$$f'_{\mathbf{s}}(\mathbf{x}) = \lim_{\lambda \to 0} \frac{f(\mathbf{x} + \lambda \mathbf{s}) - f(\mathbf{x})}{\lambda}. \tag{1.4}$$

Dá se snadno ukázat (viz například Sikorski: *Diferenciální a integrální počet funkcí více proměnných*) [1], že pak

$$f'_{\mathbf{s}}(\mathbf{x}) = \nabla f(\mathbf{x})^{\mathrm{T}} \mathbf{s} = \mathbf{s}^{\mathrm{T}} \nabla f(\mathbf{x}). \tag{1.5}$$

V „operátorovém“ vyjádření (tj. sčítání/násobení bude nyní znamenat sčítání/násobení v prostoru reálných funkcí) můžeme (1.5) stručně zapsat jako

$$f'_{\mathbf{s}} = \nabla f^{\mathrm{T}} \mathbf{s} = \mathbf{s}^{\mathrm{T}} \nabla f. \tag{1.6}$$

Průběh funkce f na přímce $\mathbf{x}(\alpha) = \mathbf{x}' + \alpha \mathbf{s}$ (viz (1.1)) je dán jednorozměrnou funkcí $\phi(\alpha) = f(\mathbf{x}(\alpha))$ a platí

$$f'_{\mathbf{s}}(\mathbf{x}(\alpha)) = \phi'(\alpha). \tag{1.7}$$

Obdobně se dá ukázat, že

$$f''_{\mathbf{s},\mathbf{s}} \stackrel{\mathrm{def}}{=} (f'_{\mathbf{s}})'_{\mathbf{s}} = \mathbf{s}^{\mathrm{T}} (\nabla^2 f) \mathbf{s}. \tag{1.8}$$

Opět

$$f''_{\mathbf{s},\mathbf{s}}(\mathbf{x}(\alpha)) = \phi''(\alpha).$$

Výraz

$$\mathbf{s}^{\mathrm{T}} \nabla f(\mathbf{x}) \tag{1.9}$$

udává *sklon* funkce f podél přímky $\mathbf{x}(\alpha)$ a

$$\mathbf{s}^{\mathrm{T}} \nabla^2 f(\mathbf{x}) \mathbf{s} \tag{1.10}$$

[1] Důkaz plyne z rovnic $f'_{c\mathbf{s}} = c f'_{\mathbf{s}}$ a $f'_{\mathbf{s}+\mathbf{t}} = f'_{\mathbf{s}} + f'_{\mathbf{t}}$.

udává *zakřivení* f podle $\mathbf{x}(\alpha)$. Protože ale tyto definice jsou závislé na *délce* vektoru $\mathbf{s}$ (tj. na měřítku), je kvůli jednoznačnosti někdy potřebné připojit předpoklad $\|\mathbf{s}\| = 1$, kde $\|.\|$ je jistá pevně zvolená norma (to ale nebude nutné v případech, kdy se nám bude jednat jen o *znaménko* sklonu, resp. zakřivení).

Funkce $f : \mathbb{R}^n \to \mathbb{R}^m$ je *lipschitzovsky spojitá*, pokud existuje $\lambda > 0$ tak, že pro každé $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ je

$$\|f(x) - f(y)\| \leq \lambda \|\mathbf{x} - \mathbf{y}\|. \tag{1.11}$$

Každá lipschitzovsky spojitá funkce je zřejmě spojitá. Naopak má-li funkce f ohraničené všechny první parciální derivace, pak je lipschitzovsky spojitá.

Jako *lineární funkci* budeme označovat každou funkci tvaru

$$l(\mathbf{x}) = \sum_{i=1}^{n} a_i x_i + b = \mathbf{a}^{\mathrm{T}}\mathbf{x} + b, \tag{1.12}$$

kde $\mathbf{a} \in \mathbb{R}^n$ a $b \in \mathbb{R}$ jsou konstanty. Obecnou *kvadratickou funkci* můžeme zapsat ve tvaru

$$q(\mathbf{x}) = \tfrac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{G}\mathbf{x} + \mathbf{b}^{\mathrm{T}}\mathbf{x} + c \tag{1.13}$$

kde $\mathbf{G}$, $\mathbf{b}$ a c jsou konstantní a matice $\mathbf{G}$ je symetrická. Jsou-li $\mathbf{u}$ a $\mathbf{v}$ vektory závislé na proměnné $\mathbf{x}$ (tedy $\mathbf{u}, \mathbf{v} : \mathbb{R}^n \to \mathbb{R}^m$), pak z pravidel pro derivování součinu funkcí plyne

$$\nabla(\mathbf{u}^{\mathrm{T}}\mathbf{v}) = (\nabla\mathbf{u}^{\mathrm{T}})\mathbf{v} + (\nabla\mathbf{v}^{\mathrm{T}})\mathbf{u}. \tag{1.14}$$

Položíme-li tedy v (1.13) $\mathbf{u} = \mathbf{x}$ a $\mathbf{v} = \mathbf{G}\mathbf{x}$, dostáváme

$$\nabla q(\mathbf{x}) = \tfrac{1}{2}(\mathbf{G} + \mathbf{G}^{\mathrm{T}})\mathbf{x} + \mathbf{b} = \mathbf{G}\mathbf{x} + \mathbf{b} \tag{1.15}$$

(zde jsme využili symetrie matice $\mathbf{G}$). Obdobně se dá odvodit, že $\nabla^2 q = \mathbf{G}$. Tedy Hessova matice kvadratické funkce (1.13) je rovna (konstantní) matici $\mathbf{G}$ a její gradient je lineární funkce.

Jsou-li $\mathbf{x}$ a $\mathbf{x}'$ dva dané body, pak

$$\nabla q(\mathbf{x}') - \nabla q(\mathbf{x}) = \mathbf{G}(\mathbf{x}' - \mathbf{x}). \tag{1.16}$$

Tedy u kvadratických funkcí (1.13) převádí matice $\mathbf{G}$ změnu v argumentu funkce na změnu gradientu.

Taylorův rozvoj pro gradient můžeme zapsat ve tvaru

$$\nabla f(\mathbf{x} + \mathbf{h}) = \nabla f(\mathbf{x}) + \left[\nabla^2 f(\mathbf{x})\right] \mathbf{h} + \cdots \tag{1.17}$$

Lze tedy říci, že v dostatečně malém okolí bodu $\mathbf{x}$ se každá spojitě dvakrát diferencovatelná funkce blíží kvadratické funkci.

1.3 Literatura

Hlavním pramenem pro přednášku je učebnice Rogera Fletchera *Practical Methods of Optimization* (2.vydání, Wiley c1987, 436 stran). Tato kniha je zaměřena na metody, které byly úspěšně použity při řešení praktických úloh a zpravidla také implementovány v rámci nějakého profesionálního balíku. Text je psán volnějším stylem, tj. obvykle nesplňuje klasické matematické schéma definice–věta–důkaz.

Další významnou učebnicí, kterou také budeme částečně využívat, je *Nonlinear Programming. Theory and Algorithms* od trojice autorů M.S. Bazaraa, H.D. Sherali a C.M. Shetty (2.vydání, Wiley c1993, 638 stran). Zde je možné najít obecnější teorii matematické optimalizace a rozsáhlý přehled metod známých do roku vydání. Je to možná nejlepší úvod do oboru pro matematicky zaměřené čtenáře.

Velmi hutným přehledem výsledků do roku 1989 je monografie s názvem *Optimization*, kterou sestavili G.L. Nemhauser a A.H.G. Rinnooy Kan (North-Holland, c1989, 709 stran). Na práci se podílelo celkem 18 autorů, z velké části patřících ke klasikům a kniha patří k nejcitovanějším pramenům v oboru.

Z nedávné doby pochází útlá knížka *Optimization Software Guide*, kterou napsali J.J. Moré a S.J. Wright (SIAM, c1993, 154 stran). Jde o prvního univerzálního průvodce po existujících balících zaměřených buď speciálně na optimalizaci nebo obsahujících optimalizační procedury. Publikace má dvě části: velmi stručný přehled metod a vlastní referenční část popisující konkrétní softwarové balíky. Referenční část je dostupná také na `http://www.mcs.anl.gov` a je průběžně aktualizována.

Snadno přístupným úvodním textem je „*Mathematical Optimization*“ tvořící jednu z kapitol osvětového projektu „Computational Science Education“, viz například `http://csep1.phy.ornl.gov/csep.html`. Obsahuje i kapitoly o metodách globální optimalizaci.

2 Hledání minima funkce

Metody pro spojitou minimalizaci přirozeně vycházejí ze základních výsledků matematické analýzy, které se týkají popisu lokálních minim. V této části proto nejdříve tyto výsledky stručně zopakujeme. Dále se budeme zabývat konvergencí minimalizačních metod. Seznámíme se také se dvěma jednoduchými minimalizačními metodami: metodou největšího spádu a Nelder-Meadovou metodou. Obě metody se snadno implementují a proto jsou dost oblíbené. Jejich nevýhodou je ale velmi pomalá konvergence pro některé typy funkcí a proto pro praktické výpočty mají význam jen tehdy, nemáme-li k dispozici hotovou implementaci dokonalejších metod. Nicméně je užitečné se s nimi seznámit, protože reprezentují jisté základní přístupy v čisté formě.

2.1 Nutné podmínky

Řekneme, že bod $\mathbf{x}^*$ je *lokálním minimem* (nebo jednoduše pouze *minimem*) funkce f, pokud pro každý bod $\mathbf{x}$, který leží dostatečně blízko k $\mathbf{x}$ platí $f(\mathbf{x}^*) \leq f(\mathbf{x})$. Pokud $f(\mathbf{x}^*) \leq f(\mathbf{x})$ pro všechna $\mathbf{x}$ z definičního oboru funkce f, pak je $\mathbf{x}^*$ *globálním minimem* funkce f. Obdobně je definováno lokální a globální maximum funkce f.

Může se stát, že při takto definovaných minimech a maximech dostaneme pro některé funkce „paradoxní“ výsledky. Například $x = 0$ je lokálním minimem funkce $f(x) = \min(1 + x, 0, 1 - x)$, ale zároveň je také globálním maximem funkce f. K tomu, abychom mohli odlišit podobné krajní případy definujeme *ostré lokální minimum* (a obdobně maximum) podmínkou $f(\mathbf{x}) < f(\mathbf{x}^*)$ pro všechna $\mathbf{x}$, $\mathbf{x} \neq \mathbf{x}^*$ z jistého okolí bodu $\mathbf{x}^*$. Existují „patologické“ funkce, které mají v nějaké omezené oblasti nekonečně mnoho minim. Například to platí pro funkci $f(x) = x^2(1+x^2+\sin(1/x))$, kde $f(0) \stackrel{\text{def}}{=} 0$, v (libovolně malém) okolí bodu 0. Proto definujeme ještě *izolované lokální minimum*, kdy $\mathbf{x}^*$ musí být *jediným* lokálním minimem v jistém svém okolí.

Snadno se dá ukázat, že každé lokální minimum $\mathbf{x}^*$ dvakrát spojitě diferencovatelné funkce f musí splňovat následující dvě podmínky:

$$\mathbf{g}^* = \mathbf{0}, \tag{2.1}$$

kde $\mathbf{g}^*$ je gradient funkce f v bodě $\mathbf{x}^*$ a

$$\mathbf{s}^{\mathrm{T}}\mathbf{G}^*\mathbf{s} \geq 0, \quad \forall \mathbf{s}, \tag{2.2}$$

kde $\mathbf{G}^*$ je hessián f v bodě $\mathbf{x}^*$. Oba výsledky se dají snadno odvodit například z rovnic (1.5) a (1.6).

Pokud bod $\mathbf{x}^*$ splňuje podmínku (2.1), budeme o něm mluvit jako o *stacionárním bodě*. Mezi stacionární body patří nejen lokální minima, ale také lokální maxima, sedlové body nebo inflexní body.

2.2 Dostatečné podmínky

Abychom mohli jednoduše zformulovat a dokázat větu stanovící dostatečné podmínky pro lokální minimum, musíme nejprve uvést definici a základní tvrzení týkající se kladně definitních matic. Ta se nám ale budou hodit i v dalším výkladu.

Je-li $\mathbf{A}$ reálná symetrická matice, pak řekneme, že $\mathbf{A}$ je *kladně definitní*, pokud

$$\mathbf{s}^{\mathrm{T}}\mathbf{A}\mathbf{s} > 0, \quad \forall \mathbf{s} \neq \mathbf{0}. \tag{2.3}$$

Matice $\mathbf{A}$ je *kladně semidefinitní*, pokud je výše uvedená nerovnost neostrá, tj. pokud $\mathbf{s}^{\mathrm{T}}\mathbf{A}\mathbf{s} \geq 0, \quad \forall \mathbf{s}$. Obdobně řekneme, že $\mathbf{A}$ je *negativně definitní (semidefinitní)*, pokud $\mathbf{s}^{\mathrm{T}}\mathbf{A}\mathbf{s} < 0$ ($\mathbf{s}^{\mathrm{T}}\mathbf{A}\mathbf{s} \leq 0$) pro všechna $\mathbf{s}$.

Vlastní hodnota komplexní čtvercové matice $\mathbf{A}$ je takové komplexní číslo λ, pro které existuje komplexní vektor $\mathbf{x}$ tak, že

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}. \tag{2.4}$$

Je-li $\mathbf{A}$ reálná a symetrická, pak jsou všechny její vlastní hodnoty reálné.

Dá se ukázat, že matice je kladně definitní právě když jsou všechny její vlastní hodnoty kladné. Tohoto faktu využijeme při důkazu následující věty, kde budeme potřebovat tvrzení, že pro každou kladně definitní matici $\mathbf{G}$ existuje $\alpha > 0$ tak, že platí

$$\boldsymbol{\delta}^{\mathrm{T}}\mathbf{G}\boldsymbol{\delta} \geq a\boldsymbol{\delta}^{\mathrm{T}}\boldsymbol{\delta} \quad \forall \boldsymbol{\delta}. \tag{2.5}$$

Položme $D = \{\mathbf{x} \in \mathbb{R}^n;\ \|\mathbf{x}\| = 1\}$. Protože $q(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{G}\mathbf{x}$ je spojitá funkce a protože manožina D je uzavřená a omezená, existuje minimum funkce q na D. Položme $a = \min_D q(\mathbf{x})$. Protože pro všechna $\mathbf{x} \neq \mathbf{0}$ je $q(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{G}\mathbf{x} > 0$, je také $a > 0$. Pro libovolný vektor $\boldsymbol{\delta} \neq \mathbf{0}$ pak zřejmě $\frac{\boldsymbol{\delta}^{\mathrm{T}}}{\|\boldsymbol{\delta}\|}\mathbf{G}\frac{\boldsymbol{\delta}}{\|\boldsymbol{\delta}\|} \geq a$, odkud dostáváme naše tvrzení. Dá se navíc dokázat, že a je nejmenší vlastní hodnota matice $\mathbf{G}$ [2].

Zavedeme ještě takzvanou „o-notaci“, která podstatně zjednodušuje zápis asymptotických vyjádření funkcí, jako je například Taylorova řada. Je-li $g : \mathbb{R}^n \to \mathbb{R}$ reálná funkce, pak symbolem $o(g(\boldsymbol{\varepsilon}))$ budeme rozumět množinu všech n-rozměrných reálných funkcí f takových, že

$$\lim_{\boldsymbol{\varepsilon} \to \mathbf{0}} \frac{f(\boldsymbol{\varepsilon})}{g(\boldsymbol{\varepsilon})} = 0. \tag{2.6}$$

Je-li tedy $f \in o(g)$, pak se dá říci, že pro $\boldsymbol{\varepsilon} \to \mathbf{0}$ klesá funkce f podstatně rychleji než funkce g. Namísto zápisu $f \in o(g)$ se běžně používá zápis $f = o(g)$. To má svoje oprávnění, protože s takovými „rovnicemi“ lze korektně provádět všechny obvyklé úpravy, za předpokladu, že všechny o-výrazy zůstanou na pravé straně a příslušné limity existují. Speciálním případem je $o(\mathbf{1})$ $(\mathbf{1}(\mathbf{x}) = 1 \quad \forall \mathbf{x})$, což je jednoduše množina funkcí, které jdou k 0 pro $\boldsymbol{\varepsilon} \to \mathbf{0}$.

Pokud se v rovnici s o-notací vyskytuje více proměnných je symbol $o(\mathbf{1})$ nejednoznačný. Například v rovnici

$$f(\varepsilon, \delta) = g(\varepsilon, \delta) + o(\mathbf{1})$$

není bez znalosti kontextu jasné, jestli se g blíží f pro $\varepsilon \to 0$, nebo pro $\delta \to 0$, nebo snad pro $(\varepsilon, \delta) \to (0, 0)$. V takovém případě budeme používat značení tvaru $o_{\boldsymbol{\varepsilon}}(\mathbf{1})$, což bude množina funkcí, které jdou k nule pro $\boldsymbol{\varepsilon} \to \mathbf{0}$ [3].

Věta 2.1. *Nechť funkce f je dvakrát spojitě diferencovatelná. Pokud $\mathbf{x}^*$ je stacionárním bodem funkce f a matice $\mathbf{G}^* = \nabla^2 f(\mathbf{x}^*)$ je kladně definitní, pak $\mathbf{x}^*$ je ostrým a izolovaným minimem funkce f.*

---

[2] Důkaz plyne z věty podle které existuje ortogonální matice $\mathbf{Q}$ taková, že $\mathbf{Q}^{\mathrm{T}}\mathbf{G}\mathbf{Q}$ je diagonální matice, která má na diagonále vlastní hodnoty matice $\mathbf{Q}$. Tvrzení pak plyne z faktu, že ortogonální transformace zachovává délky a tedy platí $D = \mathbf{Q}D = \{\mathbf{Q}\mathbf{x};\ \mathbf{x} \in D\}$.

[3] Pokud bychom chtěli být zcela přesní podle zvyklostí moderní matematické analýzy, měli bychom symbol $\boldsymbol{\varepsilon}$ v $o_{\boldsymbol{\varepsilon}}(\mathbf{1})$ definovat jako seznam indexů proměnných. Uvedená definice je ale srozumitelná a umožňuje vyhnout se komplikovaným formulacím.

Důkaz. Uvažujme libovolný bod $\mathbf{x}^* + \boldsymbol{\delta}$, $\boldsymbol{\delta} \neq \mathbf{0}$. Taylorův rozvoj funkce $f(\mathbf{x}^* + \boldsymbol{\delta})$ se dá zapsat jako

$$f(\mathbf{x}^* + \boldsymbol{\delta}) = f^* + \boldsymbol{\delta}^{\mathrm{T}}\mathbf{g}^* + \tfrac{1}{2}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{G}^*\boldsymbol{\delta} + o(\boldsymbol{\delta}^{\mathrm{T}}\boldsymbol{\delta})$$

(kde $f^* = f(\mathbf{x}^*)$). Z (2.1) dostáváme

$$f(\mathbf{x}^* + \boldsymbol{\delta}) = f^* + \tfrac{1}{2}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{G}^*\boldsymbol{\delta} + o(\boldsymbol{\delta}^{\mathrm{T}}\boldsymbol{\delta}).$$

Podle (2.5) platí $\boldsymbol{\delta}^{\mathrm{T}}\mathbf{G}^*\boldsymbol{\delta} \geq a\boldsymbol{\delta}^{\mathrm{T}}\boldsymbol{\delta}$ a tedy

$$f(\mathbf{x}^* + \boldsymbol{\delta}) \geq f^* + \left[\tfrac{1}{2}a + o_{\boldsymbol{\delta}}(\mathbf{1})\right]\boldsymbol{\delta}^{\mathrm{T}}\boldsymbol{\delta}.$$

Jde-li $\boldsymbol{\delta}$ k $\mathbf{0}$, pak $o_{\boldsymbol{\delta}}(\mathbf{1}) \to 0$. Protože $a > 0$ je pevné, musí pro $\boldsymbol{\delta}$ dostatečně blízké k $\mathbf{0}$ být $\frac{1}{2}a > |o_{\boldsymbol{\delta}}(\mathbf{1})|$. Tedy $f(\mathbf{x}^* + \boldsymbol{\delta}) > f^*$ a $\mathbf{x}^*$ je ostré lokální minimum.

Uvažujme nyní posloupnost bodů $\mathbf{x}^* + \boldsymbol{\delta}^{(i)}$, které jsou lokálními minimy funkce f a pro které $\boldsymbol{\delta}^{(i)} \to \mathbf{0}$ pro $i \to \infty$ ($\boldsymbol{\delta}^{(i)} \neq \mathbf{0}$). Z Taylorova rozvoje pro funkci $\mathbf{g}$ (rovnice (1.17)) plyne

$$\boldsymbol{\delta}^{\mathrm{T}}\mathbf{g}(\mathbf{x}^* + \boldsymbol{\delta}) = \boldsymbol{\delta}^{\mathrm{T}}\mathbf{g}^* + \boldsymbol{\delta}^{\mathrm{T}}\mathbf{G}^*\boldsymbol{\delta} + o(\|\boldsymbol{\delta}\|^2).$$

Podle předpokladů je $\mathbf{g}^* = \mathbf{g}(\mathbf{x}^* + \boldsymbol{\delta}^{(i)}) = \mathbf{0}$. Proto

$$0 = \boldsymbol{\delta}^{(i)\mathrm{T}}\mathbf{G}^*\boldsymbol{\delta}^{(i)} + o(\|\boldsymbol{\delta}^{(i)}\|^2) \qquad \forall i.$$

Protože $\boldsymbol{\delta}^{(i)\mathrm{T}}\mathbf{G}^*\boldsymbol{\delta}^{(i)} \geq a\boldsymbol{\delta}^{(i)\mathrm{T}}\boldsymbol{\delta}^{(i)}$, můžeme pak obdobně jako v první části důkazu psát

$$0 \geq \left[a + o_{\boldsymbol{\delta}^{(i)}}(\mathbf{1})\right]\boldsymbol{\delta}^{(i)\mathrm{T}}\boldsymbol{\delta}^{(i)} \qquad \forall i$$

(neboť $\|\boldsymbol{\delta}^{(i)}\|^2 = \boldsymbol{\delta}^{(i)\mathrm{T}}\boldsymbol{\delta}^{(i)}$). To ale pro dostatečně velké i (tj. pro $\boldsymbol{\delta}^{(i)}$ dostatečně blízké $\mathbf{0}$) nemůže platit, protože pak $a > \left|o_{\boldsymbol{\delta}^{(i)}}(\mathbf{1})\right|$ a tedy $a + o_{\boldsymbol{\delta}^{(i)}}(\mathbf{1}) > 0$. Tedy $\mathbf{x}^*$ je izolované lokální minimum. □

2.3 Nelder-Meadova metoda

Základní myšlenkou Nelder-Meadovy metody je představa, že se n-rozměrným prostorem pohybuje jistý objekt („améba“), který se může natahovat nebo zkracovat v různých směrech. Několik typů takových transformací má zajistit, aby se objekt posouval směrem do „údolí“ a po dosažení dna údolí se „plazil“ co nejkratší cestou k lokálnímu minimu.

Zmíněný objekt je definován jako simplex s $n+1$ vrcholy s neprázdným obsahem, tj. jde o konvexní obal $n+1$ bodů. Ve třírozměrném prostoru je to čtyřstěn. Někdy je tato metoda označována jako simplexová, to ale vede často k nedorozumění, protože simplexovou metodou se zpravidla míní známá metoda pro lineární optimalizaci.

Používají se následující transformace:

- Reflexe: bod $\mathbf{p}_i$, který má největší funkční hodnotu se přemístí (odzrcadlí) na druhou stranu simplexu, tj. k bodu $\mathbf{p}_i$ se přičte dvojnásobek rozdílu mezi $\mathbf{p}_i$ a průměrem ostatních bodů ($\sum_{j\neq i}\mathbf{p}_j/n$).

- Reflexe a prodloužení: totéž jako v předchozím případě, až na to, že simplex je prodloužen v novém směru (tj. přičítá se více než dvojnásobek rozdílu mezi nejhorším bodem a průměrem ostatních).

- Kontrakce: nejhorší bod se přibíží k průměru ostatních. To je vhodné zejména v případě, kdy má „améba“ projít úzkým údolím.

Na začátku se simplex zpravidla definuje jako

$$\mathbf{p}_i = \mathbf{p}_0 + \lambda \mathbf{e}_i \quad (i = 1, \ldots, n),$$

kde $\mathbf{p}_0$ je pevně zvolený bod, $\mathbf{e}_i$ jsou jednotkové vektory a λ je konstanta odrážející odhad měřítka optimalizačního problému (např. šířku „údolí“).

Metoda obvykle končí poté, co není dosaženo výrazného zlepšení účelové funkce nebo pokud se v některém cyklu simplex téměř nezmění.

Nevýhodou Nelder-Meadovy metody je pomalá konvergence, přičemž se může dokonce stát, že výpočet neskončí v lokálním minimu.

2.4 Metoda největšího spádu

Zdálo by se, že nejlepším způsobem jak dospět k lokálnímu minimu, je vydat se směrem, ve kterém účelová funkce nejrychleji klesá, tj. ve směru $-\nabla f(\mathbf{x})$, kde $\mathbf{x}$ je bod, ve kterém se výpočet právě nachází. Z této myšlenky vychází *metoda největšího spádu*:

- Vyjdi ze zvoleného bodu $\mathbf{x}^{(0)}$.
- V i-té iteraci polož $\mathbf{x}^{(i)} = \mathbf{x}^{(i-1)} - \alpha^* \nabla f(\mathbf{x}^{(i-1)})$, kde α^* určuje minimum funkce f na polopřímce $\mathbf{x}^{(i-1)} - \alpha \nabla f(\mathbf{x}^{(i-1)})$, $\alpha > 0$.

Není obtížné ukázat, že pro každé i je ${\mathbf{g}^{(i-1)}}^{\mathrm{T}} \mathbf{g}^{(i)} = 0$ (kde $\mathbf{g}^{(i)} = \nabla f(\mathbf{x}^{(i)})$) a tedy po sobě jdoucí směry minimalizace jsou na sebe *kolmé* [4]. Takový postup ale obecně *nevede* k rychlému nalezení minima. Intuitivně si lze metodu největšího spádu představit jako lyžaře, který se snaží dostat na dno úzkého údolí tak, že vždy z místa, na kterém právě je namíří ve směru největšího spádu. Pro náhodně vybraný počáteční bod ale pravděpodobně bude muset mnohokrát přejet z jedné strany údolí na druhou, tj. počet „iterací“ bude neúměrně velký. Tento problém se dá samozřejmě demonstrovat i na formálně zadaných funkcích dvou proměnných.

Dalším nedostatkem metody největšího spádu, který již není tak dobře viditelný, jsou chyby způsobené zaokrouhlováním. Mohou vést i k tomu, že se výpočet vůbec nedostane rozumně blízko k minimu. Na druhé straně se ale dá ukázat, že při (ideální) přesné aritmetice metoda konverguje vždy k nějakému lokálnímu minimu.

2.5 Metody spádových směrů

Zobecněním metody největšího spádu je schéma metod spádových směrů:

(a) Urči směr $\mathbf{s}^{(k)}$, ve kterém bude hledán další bod.

(b) Najdi $\alpha^{(k)}$, které minimalizuje $f(\mathbf{x}^{(k)} + \alpha \mathbf{s}^{(k)})$ vzhledem k α. (2.7)

(c) Polož $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)} \mathbf{s}^{(k)}$.

[4] V minimu je $\phi'(\alpha) = f'_{\mathbf{g}^{(i-1)}}(\mathbf{x}^{(i-1)} - \alpha \mathbf{g}^{(i-1)}) = 0$. Přitom $f'_{\mathbf{g}^{(i-1)}} = (\nabla f)^{\mathrm{T}} \mathbf{g}^{(i-1)}$ podle (1.6).

Různé metody se liší tím, jak vybírají směr $\mathbf{s}^{(k)}$ v bodě (a). Bod (b), neboli *výběr délky kroku* je jednorozměrnou minimalizací f podél přímky $\mathbf{x}^{(k)}+\alpha\mathbf{s}^{(k)}$. Pokud daná minimalizační metoda používá informace o derivacích funkce f k určení $\mathbf{s}^{(k)}$, pak zpravidla derivací využívá i při určování délky kroku.

Krok (b) v (2.7) je idealizovaný: obecně není možné najít *přesné* minimum v konečném počtu kroků. Dokonce je vhodnější tuto jednorozměrnou minimalizaci nedělat *příliš* přesně, tedy zejména ne s největší přesností dosažitelnou v pohyblivé řádové čárce na použitém počítači. Ukazuje se, že pro běžně používané metody se i pro značně volný odhad minima funkce $\phi(\alpha) = f(\mathbf{x}^{(k)} + \alpha\mathbf{s}^{(k)})$ počet iterací znatelně nezvýší. Tím se samozřejmě výpočet značně urychluje, protože pro nepřesné určení minima stačí daleko menší počet vyhodnocení funkce f a případně jejích derivací. Algoritmy pro přibližné určení minima v bodě (2.7b) se budeme podrobněji zabývat v odstavci 2.6.

Smyslem každé iterace je samozřejmě snížení hodnoty minimalizované funkce. Aby takové snížení bylo v (2.7) zaručeno, je nutné, aby funkce f ve směru $\mathbf{s}^{(k)}$ klesala (bez újmy na obecnosti můžeme předpokládat hledání minima pouze ve směru $\alpha > 0$). Pak řekneme, že vektor $\mathbf{s}^{(k)}$ je *spádovým směrem*. Podle (1.5) to znamená, že

$$\mathbf{s}^{(k)\mathrm{T}}\mathbf{g}^{(k)} < 0. \tag{2.8}$$

Nejjednodušší metodou spádových směrů je metoda největšího spádu, ve které $\mathbf{s}^{(k)} = -\mathbf{g}^{(k)}$ pro všechna k.

Otázkou, kterou je třeba řešit při každém reálném optimalizačním výpočtu, je stanovení *podmínky ukončení*. Protože v minimu je $\nabla f = \mathbf{g} = \mathbf{0}$, jednou z přirozených možností je

$$\|\mathbf{g}^{(k)}\| \leq \varepsilon. \tag{2.9}$$

Nicméně v některých ohledech není (2.9) ideálním kritériem. Například nezohledňuje možnost, kdy je minimalizační úloha špatně podmíněná, tj. kdy v některých směrech je prohledávané „údolí“ daleko užší než v jiných.[5] Pokud očekáváme, že optimalizační algoritmus bude v blízkém okolí minima rychle konvergovat, bývá výhodnější použít kriterium založené na velikosti posunutí jednotlivých proměnných v k-té iteraci:

$$\|x_i^{(k)} - x_i^{(k+1)}\| \leq \varepsilon_i \quad \forall i, \tag{2.10}$$

nebo na velikosti snížení funkční hodnoty:

$$f^{(k)} - f^{(k+1)} \leq \varepsilon. \tag{2.11}$$

Z praktických zkušeností vyplývá, že pro newtonovské metody dobře funguje test (2.11). Pro metody, které v blízkosti minima konvergují pomaleji, jako je metoda sdružených gradientů, obvykle lépe vyhovuje test (2.9). Oběma zmíněnými typy metod se budeme později podrobně zabývat.

Metoda největšího spádu může pro špatně podmíněné problémy konvergovat extrémně pomalu, a to dokonce i tehdy, když se omezíme na minimalizaci kvadratických funkcí (definovaných podle (1.13)). Nabízí se otázka, jak tedy lépe zvolit spádové směry, aby minimalizace

[5] Přesněji řečeno, měřítkem špatné podmíněnosti optimalizační úlohy je podíl největší a nejmenší vlastní hodnoty Hessovy matice funkce f v oblasti $\{\mathbf{x};\ f(\mathbf{x}) \leq f(\mathbf{x}^{(1)})\}$ (předpokládáme, že na této oblasti je Hessova matice kladně definitní).

byla efektivní přinejmenším pro kvadratické funkce. Následující věta ukazuje, že je žádoucí, aby spádové směry $\mathbf{s}^{(1)}, \mathbf{s}^{(2)}, \ldots, \mathbf{s}^{(n)}$ byly *sdružené* vzhledem k matici $\mathbf{G}$ z (1.13) (to je zároveň Hessova matice dané kvadratické funkce na $\mathbb{R}^n$), tj. aby

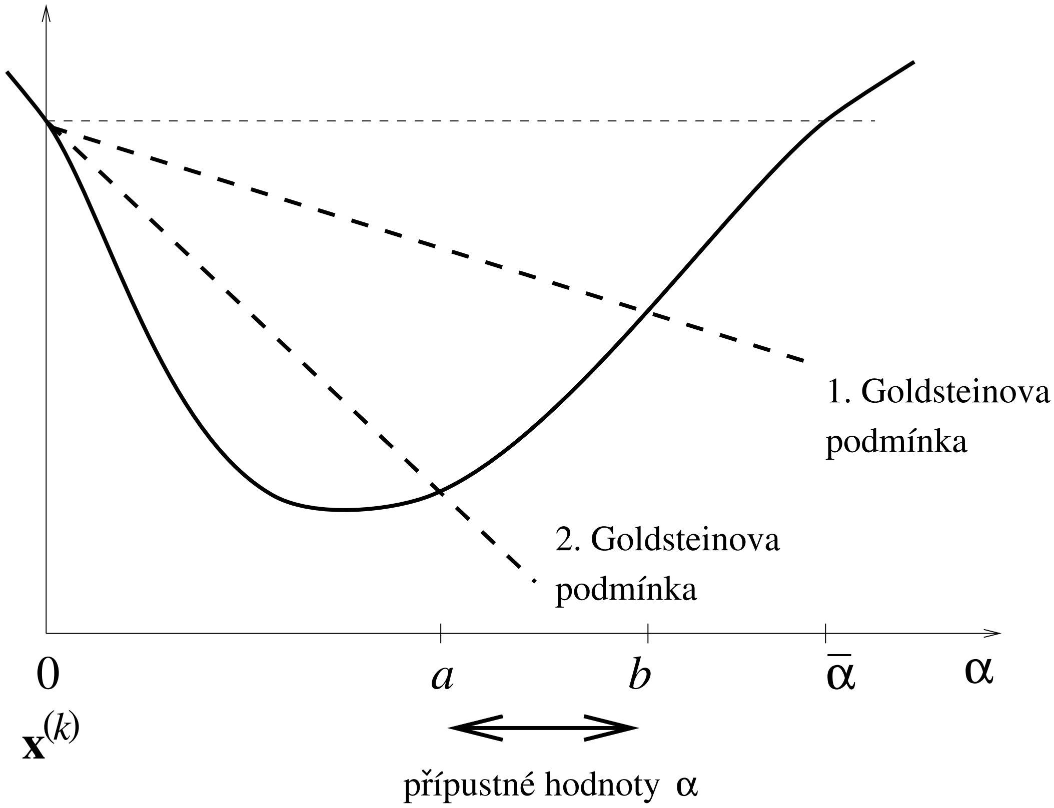
$$\mathbf{s}^{(i)^{\mathrm{T}}}\mathbf{G}\mathbf{s}^{(j)} = 0 \quad \forall i \neq j. \tag{2.12}$$

Metoda, pro kterou platí (2.12) nazveme *metodou sdružených směrů.*

Věta 2.2. *Metoda sdružených směrů najde minimum kvadratické funkce f o n proměnných nejvýše po n iteracích s přesným hledáním délky kroku. Každý bod $\mathbf{x}^{(k+1)}$ (vzniklý v k-té iteraci metody) je pak minimem funkce f v podprostoru generovaném vektory $\mathbf{s}^{(1)}, \mathbf{s}^{(2)}, \ldots, \mathbf{s}^{(k)}$ (jde o množinu*
$\{\mathbf{x};\, \mathbf{x} = \mathbf{x}^{(1)} + \sum_{j=1}^{k} \alpha_j \mathbf{s}^{(j)},\, \alpha_j \in \mathbb{R}\}$).

2.6 Výběr délky kroku

V algoritmu (2.7) je v kroku (b) formulován maximalistický požadavek na nalezení přesného minima funkce f podél směru $\mathbf{s}^{(k)}$. Z důvodu efektivnosti je ale lepší tento požadavek oslabit a volit α tak, aby v podstatě pouze $f^{(k+1)} < f^{(k)}$. Samotný požadavek na úbytek hodnoty účelové funkce by ale nestačil k tomu, aby výpočet konvergoval k lokálnímu minimu. Evidentně je tedy vhodné zvolit nějaký kompromis mezi přesným vyhledáváním a slabým požadavkem $f^{(k+1)} < f^{(k)}$.



Obrázek 1: Goldsteinovy podmínky.

Předpokládejme, že funkce f má průběh naznačený na obrázku 1, kdy existuje $\alpha > 0$ tak, že $f(\mathbf{x}^{(k)} + \alpha\mathbf{s}^{(k)}) = f(\mathbf{x}^{(k)})$. Nejmenší takové α označme symbolem $\bar{\alpha}$. Dostatečné zlepšení funkční hodnoty můžeme nyní interpretovat tak, že $\alpha^{(k)}$ není příliš blízko ani k 0, ani k $\bar{\alpha}$. Vyjádřením tohoto požadavku jsou klasické *Goldsteinovy podmínky.*

První Goldsteinova podmínka zaručuje, že α nebude zvoleno příliš blízko $\bar{\alpha}$:

$$f(\mathbf{x}^{(k+1)}) = f(\mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{s}^{(k)}) \leq f(\mathbf{x}^{(k)}) + \alpha^{(k)}\rho f'_{\mathbf{s}^{(k)}}(\mathbf{x}^{(k)}). \tag{2.13}$$

Parametr ρ je zde pevně zvolené číslo z intervalu $(0, \frac{1}{2})$. Pokud $\bar{\alpha}$ neexistuje (tj. $f(\mathbf{x}^{(k)} + \alpha\mathbf{s}^{(k)}) < f(\mathbf{x}^{(k)})$ pro všechna $\alpha > 0$), ale funkce f je zdola omezená, pak podmínka (2.13) rovněž omezuje volbu $\alpha^{(k)}$.

Druhá Goldsteinova podmínka

$$f(\mathbf{x}^{(k+1)}) = f(\mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{s}^{(k)}) \geq f(\mathbf{x}^{(k)}) + \alpha^{(k)}(1-\rho) f'_{\mathbf{s}^{(k)}}(\mathbf{x}^{(k)}) \tag{2.14}$$

zaručuje, že $\alpha^{(k)}$ nebude zvoleno příliš blízko 0. Pravé strany Goldsteinových podmínek určují dvě přímky se zápornou směrnicí, neboť podle předpokladu je $\mathbf{s}^{(k)}$ spádový směr v $\mathbf{x}^{(k)}$ a tedy podle (1.5) je $f'_{\mathbf{s}^{(k)}}(\mathbf{x}^{(k)}) = \mathbf{g}(\mathbf{x}^{(k)})^{\mathrm{T}}\mathbf{s}^{(k)} < 0$. Protože $\rho \in (0, \frac{1}{2})$, existuje vždy $\alpha^{(k)}$ splňující obě podmínky.

Namísto druhé Goldsteinovy podmínky se v současné době zpravidla používá podmínka testující sklon funkce $f(\alpha) = f(\mathbf{x}^{(k)} + \alpha\mathbf{s}^{(k)})$ v bodě $\alpha^{(k)}$:

$$|f'_{\mathbf{s}^{(k)}}(\mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{s}^{(k)})| \leq \sigma |f'_{\mathbf{s}^{(k)}}(\mathbf{x}^{(k)})|, \tag{2.15}$$

kde $1 > \sigma > \rho$. Opět vždy existuje $\alpha^{(k)}$ splňující zároveň (2.13) a (2.15).

Věta 2.3. *Nechť funkce f je spojitě diferencovatelná a nechť její gradient $\mathbf{g} = \nabla f$ je lipschitzovsky spojitý na $\mathbb{R}^n$. Je-li $\{\mathbf{x}^{(k)}\}$ posloupnost generovaná metodou s nepřesnou délkou kroku splňující* (2.13) *a* (2.15) *a pokud navíc existuje $\gamma > 0$ tak, že*

$$\mathbf{g}(\mathbf{x}^{(k)})^{\mathrm{T}}\mathbf{s}^{(k)} < -\gamma \|\mathbf{g}(\mathbf{x}^{(k)})\|_2 \|\mathbf{s}^{(k)}\|_2 \quad \forall k, \tag{2.16}$$

pak buď $\mathbf{g}(\mathbf{x}^{(k)}) = 0$ pro některé k, nebo $\lim_{k\to\infty} \mathbf{g}(\mathbf{x}^{(k)}) = 0$.

Podmínka (2.16) neříká nic jiného než že úhel mezi $\mathbf{s}^{(k)}$ a negativně vzatým gradientem v bodě $\mathbf{x}^{(k)}$ je menší než nějaký pevně zvolený úhel $< 90°$. Pro parametry z (2.13) a z (2.15) jsou v typicky voleny velmi „mírné“ hodnoty: α se volí kolem 10^{-4}, takže téměř libovolně malý úbytek funkční hodnoty je akceptovatelný; σ se volí mezi 0.7 a 0.9, takže ke splnění (2.15) stačí i malé snížení sklonu funkce f.

2.7 Metoda zlatého řezu

Jednou z efektivních metod pro řešení nelineárních *rovnic* je metoda půlení. Metoda zlatého řezu se dá považovat za analogii metody půlení pro jednorozměrnou optimalizaci. Zatímco ale při hledání minima funkce vystačíme s intervalem $[a, b]$ takovým, že $f(a)f(b) < 0$, u minimalizace potřebujeme *tři* body, řekněme a, b a c, takové, že $a < b < c$ a $f(b) \leq f(a)$, $f(b) \leq f(c)$.

Stejně jako u metody půlení zvolíme nový bod x mezi a a c. Řekněme, že $b < c$. Pokud $f(b) < f(x)$ pak se novou trojicí bodů stává (a, b, x). Pokud $f(b) > f(x)$, pak je novou trojicí (b, x, c). V obou případech má funkce f v prostředním bodě nové trojice nejnižší dosud dosaženou hodnotu. Obdobně je tomu v případě, kdy $b > c$ (nakreslete si obrázek). Celý postup se opakuje dokud se rozdíl mezi krajními body dostatečně nezmenší.

Otázkou je, jak volit nový bod x tak, aby metoda rychle konvergovala bez ohledu na průběh minimalizované funkce (stejně jako je tomu u metody půlení). Řekněme, že bod b dělí interval $[a, c]$ v poměru $w : (1 - w)$, tj.

$$\frac{b-a}{c-a} = w \qquad \frac{c-b}{c-a} = (1-w). \tag{2.17}$$

Písmenem z označíme poměr rozdílu x a b k celkové délce intervalu, tj.

$$\frac{x-b}{c-a} = z. \tag{2.18}$$

Tedy pokud x leží vpravo od b, pak je $z > 0$, pokud je x vlevo od b, je $z < 0$.

Předpokládejme, že $b < x$. Pokud se novou trojicí stane (a, b, x), pak délka nového intervalu bude $(w + z)$-násobkem délky původního intervalu. Pokud novou trojicí bude (b, x, c), bude velikost nového intervalu $(1-w)$-násobkem původní délky. Chceme-li, aby algoritmus fungoval co nejlépe v nejhorším případě (opět analogie s metodou půlení), je vhodné volit z tak, aby $1 - w = w + z$, tedy

$$z = 1 - 2w. \tag{2.19}$$

Pokud $b > x$, pak délka nového intervalu je buď w-násobkem nebo $(1 - w - z)$-násobkem délky původního intervalu (z je v tomto případě záporné číslo; znázorněte si obě možnosti pro výběr nového intervalu). Položíme-li $w = 1 - w - z$, opět platí (2.19). Odtud se snadno odvodí, že $|x - c| = |b - a|$, a tedy x je symetrickým obrazem bodu b podle středu intervalu $[a, c]$. Zejména tedy x leží ve větším z intervalů $[a, b]$ a $[b, c]$.

Opět řekněme, že x leží v intervalu $[b, c]$. Za předpokladu, že stejný postup, kterým jsme odvodili umístění x byl v předcházející iteraci uplatněn i pro umístění b, můžeme dokonce říci *kde* přesně bod x leží. Měl by totiž dělit interval $[b, c]$ ve stejném poměru jako bod b dělí interval $[a, c]$ (neboť interval $[a, c]$ byl tím větším v předchozí iteraci). Pak tedy musí platit

$$\frac{x-b}{c-b} = w = \frac{b-a}{c-a} \tag{2.20}$$

$$\frac{x-b}{c-a}\,\frac{c-a}{c-b} = w \quad \text{po násobení } \tfrac{c-a}{c-a} = 1 \tag{2.21}$$

$$\frac{z}{1-w} = w \tag{2.22}$$

Poslední rovnice dává spolu s (2.19) kvadratickou rovnici $w^2 - 3w + 1 = 0$, jejíž řešení v intervalu $[0, 1]$ je

$$w = \frac{3-\sqrt{5}}{2} \approx 0.38197 \tag{2.23}$$

Jinými slovy, bod b z trojice (a, b, c) by měl dělit interval $[a, c]$ přibližně v poměru 0.38197 : 0.61803. Poměr $\frac{1-w}{w} = \frac{1+\sqrt{5}}{2} \approx 1.61803$ bývá běžně označován jako *zlatý řez*, od čehož je odvozen i název této minimalizační metody.

Obdobně se dá provést odvození pro $x \in [a, b]$ (vyjde pak $\frac{w}{1-w} = \frac{1+\sqrt{5}}{2}$).

Metoda zlatého řezu tedy umisťuje nový bod x tak, aby v uvedeném poměru dělil větší z intervalů $[a, b]$ a $[b, c]$, přičemž bod x je blíže bodu b než příslušnému krajnímu bodu (a nebo c). V případě, kdy počáteční trojice $(a^{(0)}, b^{(0)}, c^{(0)})$ nevyhovuje poměru zlatého řezu, iterace používající uvedených pravidel rychle k tomuto poměru konvergují.

Metoda zlatého řezu zaručuje, že nový interval v každé iteraci bude mít délku nejvýše rovnou 0.61803 násobku délky předchozího intervalu. To je o něco horší výsledek než u metody půlení, kde je příslušný poměr roven 1/2.

Kvůli urychlení se metoda zlatého řezu kombinuje s kvadratickou nebo kubickou interpolací (jednorozměrné) funkce f.

2.8 Rychlost konvergence

Rychlost konvergence iterativních metod můžeme srovnávat pomocí jistého jejího asymptotického vyjádřením. Označme odchylku od řešení v k-té iteraci jako $\mathbf{h}^{(k)}$, tj.

$$\mathbf{h}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*. \tag{2.24}$$

Pokud $\mathbf{h}^{(k)} \to \mathbf{0}$, pak řekneme, že konvergence metody je *p-tého řádu*, když $\|\mathbf{h}^{(k+1)}\| \,/ \|\mathbf{h}^{(k)}\|^p \to a$, kde $a > 0$. Z praktického hlediska jsou důležité v podstatě jen případy $p = 1$ (*lineární konvergence*) a $p = 2$ (*kvadratická konvergence*), protože nejsou známy (a patrně ani neexistují) žádné obecně použitelné numerické metody s vyšším řádem konvergence.

Podle definice pro každou lineárně konvergentní metodu existuje konstanta $\alpha > 0$ tak, že

$$\|\mathbf{h}^{(k+1)}\| \Big/ \|\mathbf{h}^{(k)}\| \leq \alpha \quad \text{neboli} \quad \mathbf{h}^{(k+1)} = O(\|\mathbf{h}^{(k)}\|),$$

obdobně pro metodu s kvadratickou konvergencí platí

$$\|\mathbf{h}^{(k+1)}\| \Big/ \|\mathbf{h}^{(k)}\|^2 \leq \alpha \quad \text{neboli} \quad \mathbf{h}^{(k+1)} = O(\|\mathbf{h}^{(k)}\|^2).$$

Metody s lineární konvergencí jsou obvykle používány jen když je konstanta α dostatečně malá, řekněme $\alpha \leq \frac{1}{4}$.

Řada minimalizačních metod, které nejsou kvadraticky konvergentní, konvergují rychleji než jak to zaručuje lineární konvergence: podíl po sobě jdoucích chyb jde pro $k \to \infty$ k nule, tj.

$$\|\mathbf{h}^{(k+1)}\| \Big/ \|\mathbf{h}^{(k)}\| \to 0 \quad \text{neboli} \quad \mathbf{h}^{(k+1)} = o(\|\mathbf{h}^{(k)}\|).$$

O takových metodách říkáme, že mají *superlineární konvergenci*.

3 Newtonovské metody

Všechny newtonovské minimalizační metody vycházejí z Taylorova rozvoje minimalizované funkce v okolí posledního bodu:

$$f(\mathbf{x}^{(k)}+\boldsymbol{\delta}) \approx q^{(k)}(\boldsymbol{\delta}) = f(\mathbf{x}^{(k)}) + \mathbf{g}^{(k)\mathrm{T}}\boldsymbol{\delta} + \frac{1}{2}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{G}^{(k)}\boldsymbol{\delta}, \tag{3.1}$$

kde $\mathbf{G}^{(k)} = \nabla^2 f(\mathbf{x}^{(k)})$ Kvadratická funkce (3.1) se přibližně minimalizuje a výsledek se stává dalším bodem iterační metody. Pokud $\boldsymbol{\delta}$ určuje minimum $q^{(k)}(\boldsymbol{\delta})$, pak $\nabla q^{(k)}(\boldsymbol{\delta}) = \mathbf{0}$ a tedy podle (1.15) $\boldsymbol{\delta}$ splňuje *Newtonovu rovnici*:

$$\mathbf{G}^{(k)}\boldsymbol{\delta} = -\mathbf{g}^{(k)}. \tag{3.2}$$

Jednotlivé typy newtonovských metod se liší v postupu, jakým vytvářejí aproximaci (3.1). Klasická Newtonova metoda počítá matici $\mathbf{G}^{(k)}$ přesně, zatímco kvazi-newtonovské metody počítají jistou její aproximaci $\mathbf{H}^{(k)} \approx \mathbf{G}^{(k)}$.

3.1 Newtonova metoda

Klasická Newtonova metoda konstruuje aproximaci přímo pomocí hessiánu funkce f, tj. $\mathbf{G}^{(k)} = \nabla^2 f(\mathbf{x}^{(k)})$. V k-tém kroku Newtonovy metody se pak provedou následující operace:

$$\begin{aligned} &\text{(a) najdi řešení } \boldsymbol{\delta}^{(k)} \text{ rovnice } \mathbf{G}^{(k)}\boldsymbol{\delta} = -\mathbf{g}^{(k)}, \\ &\text{(b) polož } \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}^{(k)}. \end{aligned} \tag{3.3}$$

S použitím Newtonovy metody je spojeno několik problémů:

- pokud matice $\mathbf{G}^{(k)}$ není kladně definitní, pak řešení rovnice $\mathbf{G}^{(k)}\boldsymbol{\delta} = -\mathbf{g}^{(k)}$ nemusí být lokálním minimem kvadratické funkce (3.1) (a metoda tedy nemusí konvergovat k minimu funkce f),
- určení $\boldsymbol{\delta}^{(k)}$ v kroku (a) vyžaduje inverzi matice $\mathbf{G}^{(k)}$, tedy čas úměrný n^3,
- uživatel musí mít prostředek (proceduru) nejen pro výpočet $f(\mathbf{x})$ a $\nabla f(\mathbf{x})$, ale i pro výpočet $\nabla^2 f(\mathbf{x})$.

Všechny uvedené problémy jsou vyřešeny v kvazi-newtonovských metodách (odstavec 3.3).

Newtonova metoda konverguje kvadraticky v okolí minima, jak ukazuje následující věta. V praxi to znamená, že v každé iteraci se počet platných cifer odhadu řešení zhruba zdvojnásobuje.

Věta 3.1. *Předpokládejme, že funkce f má spojité druhé derivace a její hessián je lipschitzovsky spojitý* [6] *v nějakém okolí lokálního minima $\mathbf{x}^*$. Pokud $\mathbf{x}^{(k)}$ je dostatečně blízko $\mathbf{x}^*$ pro některé k a pokud $\mathbf{G}^*$ je kladně definitní, pak Newtonova metoda pro funkci f konverguje kvadraticky k $\mathbf{x}^*$.*

[6] Tedy pro nějakou maticovou normu $|||\cdot|||$ a vektorovou normu $\|\cdot\|$ existuje $\lambda > 0$ tak, že $|||\mathbf{G}(\mathbf{x}) - \mathbf{G}(\mathbf{y})||| \leq \lambda\|\mathbf{x}-\mathbf{y}\|$. V praxi můžeme volit například $|||\mathbf{A}||| = \left[\sum_{i=1}^n \sum_{j=1}^n a_{ij}^2\right]^{1/2}$, $\|\cdot\| = \|\cdot\|_2$.

Není-li k dispozici procedura pro výpočet hessiánu funkce f, lze jej aproximovat pomocí konečných diferencí z gradientu. i-tý sloupec $\boldsymbol{\gamma}_i$ hessiánu $\mathbf{G}^{(k)}$ je pak odhadnut jako

$$\boldsymbol{\gamma}_i \approx \frac{\mathbf{g}(\mathbf{x}^{(k)} + h_i \mathbf{e}_i) - \mathbf{g}^{(k)}}{h_i}, \tag{3.4}$$

kde $\mathbf{e}_i$ je i-tý jednotkový vektor a $h_i > 0$ je vhodně zvolené malé číslo.

Obdobně jako u minimalizačního problému můžeme uplatnit Taylorův rozvoj (3.1) i při *řešení soustav nelineárních rovnic*. Pokud řešíme rovnici $F(\mathbf{x}) = \mathbf{0}$, kde $F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$, pak můžeme F rozvinout podle Taylorovy řady jako

$$F(\mathbf{x}^{(k)} + \boldsymbol{\delta}) \approx F(\mathbf{x}^{(k)}) + \nabla F(\mathbf{x}^{(k)})^{\mathrm{T}} \boldsymbol{\delta}, \tag{3.5}$$

pokud zanedbáme členy druhého a vyššího stupně. Pokud požadujeme, aby $F(\mathbf{x}^{(k)} + \boldsymbol{\delta}) = \mathbf{0}$, dostáváme z (3.5) *Newtonovu metodu pro soustavy nelineárních rovnic*:

$$\nabla F(\mathbf{x}^{(k)})^{\mathrm{T}} \boldsymbol{\delta} = -F(\mathbf{x}^{(k)}). \tag{3.6}$$

Newtonova metoda je zde tedy opět dána postupem (3.3), kde $\mathbf{G}^{(k)} = \nabla F(\mathbf{x}^{(k)})$ a vektor $\mathbf{g}^{(k)}$ je nahrazen funkční hodnotou $F(\mathbf{x}^{(k)})$. V jednodimenzionálním případě dostáváme z rovnice $\nabla F(x^{(k)})\delta = -F(x^{(k)})$ známou *Newtonovu–Raphsonovu metodu* (někdy bývá také označována jako *metoda tečen*):

$$x^{(k+1)} = x^{(k)} - \frac{F(x^{(k)})}{F'(x^{(k)})}. \tag{3.7}$$

3.2 Broydenova metoda pro řešení soustav nelineárních rovnic

Metody pro optimalizaci reálných funkcí úzce souvisí s řešením soustav nelineárních rovnic: splnění rovnice $\nabla f(\mathbf{x}) = \mathbf{0}$ je, jak už jsme uvedli, nutnou podmínkou k tomu, aby bod $\mathbf{x}$ byl lokálním minimem funkce f.

Broydenova metoda, kterou dále popíšeme, je praktickou a často používanou metodou pro řešení soustav nelineárních rovnic. Navíc nám později poslouží pro srovnání s newtonovskými minimalizačními metodami.

Broydenova metoda je zobecněním metody sečen pro více dimenzí. Znamená to, že řešíme-li iterativně rovnici

$$F(\mathbf{x}) = \mathbf{0}, \tag{3.8}$$

kde $F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ je nelineární funkce, aproximujeme funkci F jistou lineární funkcí shodnou s funkcí F v posledních dvou bodech. Označíme-li tyto body $\mathbf{x}^{(k)}$ a $\mathbf{x}^{(k+1)}$, znamená to, že hledáme matici $\mathbf{B}$ takovou, že

$$F(\mathbf{x}^{(k+1)}) = F(\mathbf{x}^{(k)}) + \mathbf{B}(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}), \tag{3.9}$$

což je totéž jako

$$F(\mathbf{x}^{(k)}) = F(\mathbf{x}^{(k+1)}) + \mathbf{B}(\mathbf{x}^{(k)} - \mathbf{x}^{(k+1)}). \tag{3.10}$$

Rovnici (3.9) tedy můžeme chápat i jako začátek Taylorova rozvoje funkce F *se středem v* $\mathbf{x}^{(k+1)}$ v bodě $\mathbf{x}^{(k)}$, kde matice $\mathbf{B}$ je pak derivace F v $\mathbf{x}^{(k+1)}$. V jednodimenzionálním případě je $\mathbf{B} = b \in \mathbb{R}$ určeno jednoznačně: je to jednoduše směrnice přímky proložené body $\big(x^{(k)}, F(x^{(k)})\big)$ a $\big(x^{(k+1)}, F(x^{(k+1)})\big)$. Pokud $n > 1$, tvoří množina všech řešení $\mathbf{B}$ rovnice (3.9) prostor o dimenzi $n(n-1)$.

Označíme-li hledanou matici $\mathbf{B}^{(k+1)}$, můžeme rovnici (3.10) přepsat do tvaru

$$\mathbf{B}^{(k+1)}\boldsymbol{\delta}^{(k)} = \mathbf{y}^{(k)}, \tag{3.11}$$

kde $\boldsymbol{\delta}^{(k)} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$ a $\mathbf{y}^{(k)} = F(\mathbf{x}^{(k+1)}) - F(\mathbf{x}^{(k)})$. Při iterativním výpočtu postupně stanovíme $\mathbf{B}^{(1)}$, $\mathbf{B}^{(2)}$, … . *Broydenova metoda* je definována formulí, kterou se přechází od $\mathbf{B}^{(k)}$ k $\mathbf{B}^{(k+1)}$:

$$\mathbf{B}^{(k+1)} = \mathbf{B}^{(k)} + \frac{(\mathbf{y}^{(k)} - \mathbf{B}^{(k)}\boldsymbol{\delta}^{(k)})\boldsymbol{\delta}^{(k)\mathrm{T}}}{\boldsymbol{\delta}^{(k)\mathrm{T}}\boldsymbol{\delta}^{(k)}}. \tag{3.12}$$

V k-té iteraci se tedy obdobně jako u Newtonovy metody (3.6) nejdříve vypočítá $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}^{(k)}$ z rovnice

$$\mathbf{B}^{(k)}\boldsymbol{\delta}^{(k)} = -F(\mathbf{x}^{(k)}). \tag{3.13}$$

Poté se na základě $\mathbf{B}^{(k)}$ a $\mathbf{x}^{(k+1)}$ vypočte $\mathbf{B}^{(k+1)}$. Za $\mathbf{B}^{(1)}$ se volí aproximace derivace F v bodě $\mathbf{x}^{(0)}$, která se dá získat pomocí konečných diferencí.

Broydenova metoda je v praktických aplikacích velmi oblíbená, protože typicky vyžaduje vyhodnocení funkce F v méně bodech než modifikovaná Newtonova metoda aproximující derivace $F'(\mathbf{x}^{(k)})$ pomocí konečných diferencí.

Následující věta v podstatě říká, že Broydenova metoda je takovým zobecněním (jednorozměrné) metody sečen, které minimálně mění odhad derivace $\mathbf{B}^{(k+1)}$ ve srovnání s $\mathbf{B}^{(k)}$, pokud pro porovnání použijeme Frobeniovu normu. *Frobeniova norma* reálné matice $\mathbf{A}$ je definována jako $\|A\|_F = \left[\sum_{i=1}^n \sum_{i=1}^n a_{ij}^2\right]^{1/2}$. Je to tedy euklidovská norma vektoru, který dostaneme když napíšeme řádky matice $\mathbf{A}$ za sebe.

Věta 3.2. *Nechť* $\mathbf{B}^{(0)} \in \mathbb{R}^{n\times n}$ *je reálná čtvercová matice,* $\boldsymbol{\delta}, \mathbf{y} \in \mathbb{R}^n$, $\boldsymbol{\delta} \neq 0$. *Pak matice* $\mathbf{B}^+$ *vypočtená pomocí*

$$\mathbf{B}^+ = \mathbf{B}^{(0)} + \frac{(\mathbf{y} - \mathbf{B}^{(0)}\boldsymbol{\delta})\boldsymbol{\delta}^{\mathrm{T}}}{\boldsymbol{\delta}^{\mathrm{T}}\boldsymbol{\delta}} \tag{3.14}$$

je jediným řešením úlohy

$$\min \|\mathbf{B} - \mathbf{B}^{(0)}\|_F \quad \textit{za předpokladu} \quad \mathbf{B}\boldsymbol{\delta} = \mathbf{y}.$$

3.3 Kvazi-newtonovské metody

Pokud můžeme počítat funkční hodnoty a gradient funkce f, lze jistě hledat lokální minima řešením rovnice $\mathbf{g}(\mathbf{x}) = 0$, a to například Broydenovou metodou popsanou v předchozím odstavci. Rovnice (3.11) má pak tvar

$$\mathbf{H}^{(k+1)}\boldsymbol{\delta}^{(k)} = \mathbf{y}^{(k)}, \tag{3.15}$$

kde $\boldsymbol{\delta}^{(k)} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$ a $\mathbf{y}^{(k)} = \mathbf{g}(\mathbf{x}^{(k+1)}) - \mathbf{g}(\mathbf{x}^{(k)})$. $\mathbf{H}^{(k+1)}$ zde tedy slouží jako aproximace druhé derivace (hessiánu) funkce f v bodě $\mathbf{x}^{(k+1)}$. Uvedená rovnice definuje *kvazi-newtonovské* *metody*: tyto metody počítají aproximaci hessiánu $\mathbf{H}^{(k+1)}$ pro $k+1$ iteraci tak, aby splňovala (3.15).

Algoritmus kvazi-newtonovských metod je modifikací algoritmu Newtonovy metody (3.3) doplněnou o lineární vyhledávání a rekurentní výpočet matice $\mathbf{H}^{(k)}$ (která je nyní pouze *aproximací* hessiánu):

(a) najdi řešení $\mathbf{s}^{(k)}$ rovnice $\mathbf{H}^{(k)}\mathbf{s} = -\mathbf{g}^{(k)}$,

(b) najdi přibližné minimum funkce f ve směru $\mathbf{s}^{(k)}$ (viz odstavec 2.5) a polož $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{s}^{(k)}$, (3.16)

(c) vypočti $\mathbf{H}^{(k+1)}$ na základě $\mathbf{H}^{(k)}$ tak, aby platilo (3.15).

Matice $\mathbf{H}^{(0)}$ může být libovolná kladně definitní matice; není-li známo, že minimalizační úloha má nějakou speciální strukturu, volí se obvykle $\mathbf{H}^{(0)} = \mathbf{I}$.

Pokud bychom postupovali podle Broydenovy metody, v každém kroku bychom aproximovali minimalizovanou funkci pomocí

$$f(\mathbf{x}^{(k)} + \boldsymbol{\delta}) \approx f(\mathbf{x}^{(k)}) + \mathbf{g}(\mathbf{x}^{(k)})^{\mathrm{T}}\boldsymbol{\delta} + \tfrac{1}{2}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{B}^{(k)}\boldsymbol{\delta}.$$

Pro efektivní minimalizaci takové kvadratické funkce je ale žádoucí, aby matice $\mathbf{B}^{(k)}$ byla symetrická a kladně definitní. Tuto podmínku Broydenova metoda obecně nesplňuje a nezaručuje symetrii ani kladnou definitnost $\mathbf{B}^{(k+1)}$ za předpokladu, že $\mathbf{B}^{(k)}$ tyto vlastnosti má (nestačí tedy vhodně zvolit $\mathbf{B}^{(1)}$).

Kvazi-newtonovské metody jsou obecně konstruovány tak, aby $\mathbf{H}^{(k+1)}$ byla vždy symetrická a kladně definitní. Díky tomuto faktu a díky své efektivitě jsou kvazi-newtonovské metody nejpoužívanějšími minimalizačními metodami založenými na Taylorově větě ve smyslu rovnice (3.1).

Pokud chceme zkonstruovat efektivní metodu pro generování matic $\mathbf{H}^{(k)}$, měli bychom podobně jako u Broydenovy metody hledat rekurentní vyjádření $\mathbf{H}^{(k+1)}$ z $\mathbf{H}^{(k)}$. Výpočet $\mathbf{H}^{(k+1)}$ podle této rekurentní formule by se měl dát provést pomocí relativně malého počtu operací, což v praxi znamená časovou složitost úměrnou n^2. Navíc by matice $\mathbf{H}^{(k+1)}$ měla být symetrická, pokud tuto vlastnost má $\mathbf{H}^{(k)}$. Asi nejjednodušší formulí splňující uvedené požadavky je

$$\mathbf{H}^{(k+1)} = \mathbf{H}^{(k)} + a\mathbf{u}\mathbf{u}^{\mathrm{T}}, \tag{3.17}$$

kde $\mathbf{u}$ je jistý vektor (obecně závislý na $\mathbf{H}^{(k)}$, $\boldsymbol{\delta}^{(k)}$ a $\mathbf{y}^{(k)}$). Matice $\mathbf{u}\mathbf{u}^{\mathrm{T}}$ má řád 1 a její výpočet vyžaduje jen n^2+n násobení. Z (3.15) plyne po vynásobení obou stran rovnice (3.17) vektorem $\boldsymbol{\delta}^{(k)}$ zprava:

$$\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)} = a\mathbf{u}\mathbf{u}^{\mathrm{T}}\boldsymbol{\delta}^{(k)}. \tag{3.18}$$

Odtud je vidět, že vektor $\mathbf{u}$ je skalárním násobkem vektoru $\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)}$, neboť $a\mathbf{u}\mathbf{u}^{\mathrm{T}}\boldsymbol{\delta}^{(k)} = \mathbf{u}(a\mathbf{u}^{\mathrm{T}}\boldsymbol{\delta}^{(k)})$ a $(a\mathbf{u}^{\mathrm{T}}\boldsymbol{\delta}^{(k)}) \in \mathbb{R}$. Bez újmy na obecnosti (z hlediska směru vektoru $\mathbf{u}$) můžeme proto volit $\mathbf{u} = \mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)}$. Nyní po dosazení za první $\mathbf{u}$ v rovnici (3.18) dostáváme

$$\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)} = (\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)})a\mathbf{u}^{\mathrm{T}}\boldsymbol{\delta}^{(k)}. \tag{3.19}$$

Násobením rovnice (3.19) zleva vektorem $(\|\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)}\|_2^2)^{-1}(\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)})^{\mathrm{T}}$ získáme rovnici $1 = a\mathbf{u}^{\mathrm{T}}\boldsymbol{\delta}^{(k)}$, odkud snadno vypočítáme a. Výpočet podle (3.17) je pak dán vzorcem

$$\mathbf{H}^{(k+1)} = \mathbf{H}^{(k)} + \frac{(\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)})(\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)})^{\mathrm{T}}}{(\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)})^{\mathrm{T}}\boldsymbol{\delta}^{(k)}}. \tag{3.20}$$

Dá se ukázat, že formule (3.20) obecně nezachovává kladnou definitnost matice $\mathbf{H}$, tj. pokud je $\mathbf{H}^{(k)}$ kladně definitní, pak $\mathbf{H}^{(k+1)}$ kladně definitní být nemusí. Tedy formule prvního řádu (3.17) nedává metodu, která by byla plně uspokojivá. Je proto vhodné analyzovat formule druhého řádu dané formulí

$$\mathbf{H}^{(k+1)} = \mathbf{H}^{(k)} + a\mathbf{u}\mathbf{u}^{\mathrm{T}} + b\mathbf{v}\mathbf{v}^{\mathrm{T}}. \tag{3.21}$$

Zde již nejsou vektory $\mathbf{u}$ a $\mathbf{v}$ určeny jednoznačně. Po aplikaci (3.15) dostáváme

$$\mathbf{y}^{(k)} = \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)} + a\mathbf{u}\mathbf{u}^{\mathrm{T}}\boldsymbol{\delta}^{(k)} + b\mathbf{v}\mathbf{v}^{\mathrm{T}}\boldsymbol{\delta}^{(k)}. \tag{3.22}$$

Obdobně jako u formule prvního řádu je vidět, že jistý skalární násobek vektoru $\mathbf{u}$ a jistý skalární násobek vektoru $\mathbf{v}$ dávají dohromady vektor $(\mathbf{y}^{(k)} - \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)})$. Nabízí se proto možnost volit $\mathbf{u}$ úměrné $\mathbf{y}^{(k)}$ a $\mathbf{v}$ úměrné $\mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)}$ ($\mathbf{v}$ pak závisí jen na proměných, na kterých nezávisí $\mathbf{u}$ a naopak). Můžeme pak jistě položit $\mathbf{u} = \mathbf{y}^{(k)}$ a $\mathbf{v} = \mathbf{H}^{(k)}\boldsymbol{\delta}^{(k)}$. Obdobně jako u formule prvního řádu dostáváme z (3.22) $a\mathbf{u}^{\mathrm{T}}\boldsymbol{\delta}^{(k)} = 1$ a $b\mathbf{v}^{\mathrm{T}}\boldsymbol{\delta}^{(k)} = -1$. Odtud již po dosazení do (3.21) dostáváme (s využitím symetrie $\mathbf{H}^{(k)}$) formuli

$$\mathbf{H}^{k+1}_{\mathrm{BFGS}} = \mathbf{H} + \frac{\mathbf{y}\mathbf{y}^{\mathrm{T}}}{\mathbf{y}^{\mathrm{T}}\boldsymbol{\delta}} - \frac{\mathbf{H}\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}}{\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}\boldsymbol{\delta}} \tag{3.23}$$

(horní indexy $^{(k)}$ na pravé straně vynecháváme). Tuto metodu navrhli v roce 1970 nezávisle čtyři autoři: Broyden, Fletcher, Goldfarb a Shanno. Proto bývá označována jako *metoda BFGS*.

Jinou kvazi-newtonovskou metodu dostaneme když budeme hledat řešení vyhovující rekurentní formuli druhého řádu, ale pro *inverzi* matice $\mathbf{H}^{(k)}$. Položíme-li $\mathbf{B}^{(k)} = {\mathbf{H}^{(k)}}^{-1}$, jde o formuli

$$\mathbf{B}^{k+1} = \mathbf{B}^{(k)} + a\mathbf{u}\mathbf{u}^{\mathrm{T}} + b\mathbf{v}\mathbf{v}^{\mathrm{T}}. \tag{3.24}$$

Pokud nyní volíme $\mathbf{u} = \boldsymbol{\delta}^{(k)}$ a $\mathbf{v} = \mathbf{B}^{(k)}\mathbf{y}^{(k)}$, dostáváme obdobným postupem jako u metody BFGS vztah

$$\mathbf{B}^{k+1}_{\mathrm{DFP}} = \mathbf{B} + \frac{\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}}{\boldsymbol{\delta}^{\mathrm{T}}\mathbf{y}} - \frac{\mathbf{B}\mathbf{y}\mathbf{y}^{\mathrm{T}}\mathbf{B}}{\mathbf{y}^{\mathrm{T}}\mathbf{B}\mathbf{y}} \tag{3.25}$$

Tato metoda byla v jisté podobě obsažena v práci Davidona z roku 1959 a později explicitně popsána Fletcherem a Powellem v roce 1963. Proto bývá označována zkratkou *DFP*.

Metoda BFGS se dá vyjádřit i pomocí rekurentní formule pro inverzi matice $\mathbf{H}^{(k)}$. Není tedy nutné počítat v kroku (3.16a) inverzi matice $\mathbf{H}^{(k)}$. Při stejném označení jako u metody DFP vypadá rekurentní formule následovně:

$$\mathbf{B}^{k+1}_{\mathrm{BFGS}} = \mathbf{B} + \left(1 + \frac{\mathbf{y}^{\mathrm{T}}\mathbf{B}\mathbf{y}}{\boldsymbol{\delta}^{\mathrm{T}}\mathbf{y}}\right)\frac{\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}}{\boldsymbol{\delta}^{\mathrm{T}}\mathbf{y}} - \frac{\boldsymbol{\delta}\mathbf{y}^{\mathrm{T}}\mathbf{B} + \mathbf{B}\mathbf{y}\boldsymbol{\delta}^{\mathrm{T}}}{\boldsymbol{\delta}^{\mathrm{T}}\mathbf{y}}. \tag{3.26}$$

Není obtížné ověřit (z Cauchy–Buňakovského nerovnosti $(\mathbf{x}^{\mathrm{T}}\mathbf{y})^2 \leq \mathbf{x}^{\mathrm{T}}\mathbf{x}\mathbf{y}^{\mathrm{T}}\mathbf{y}$), že jak BFGS, tak DFP zachovávají kladnou definitnost matice $\mathbf{H}$ (resp. $\mathbf{B}$), za předpokladu, že jednorozměrná minimalizace v (3.16b) splňuje podmínku (2.15). Tedy pokud za $\mathbf{H}^{(1)}$ zvolíme libovolnou kladně definitní matici, bude $\mathbf{H}^{(k)}$ kladně definitní pro každé k.

Pro BFGS a DFP platí podobná věta o lokální konvergenci jako pro Newtonovu metodu, až na to, že konvergence není kvadratická, ale pouze superlineární (to je cena za to, že kvazi-newtonovské metody nevyužívají druhé derivace funkce f). Větší počet iterací BFGS a DFP je ale vyvážen tím, že tyto metody nemusí počítat inverzi hessiánu jako Newtonova metoda.

Věta 3.3. *Nechť funkce f splňuje předpoklady věty 3.1. Pak existují $\varepsilon, \delta > 0$ taková, že BFGS i DFP konvergují superlineárně k $\mathbf{x}^*$ z každého počátečního odhadu $\mathbf{x}^{(1)}, \mathbf{H}^{(1)}$ pro který platí $\|\mathbf{x}^* - \mathbf{x}^{(1)}\| \leq \varepsilon$ a $\|\mathbf{H}^{(1)} - \mathbf{G}^*\| \leq \delta$.*

Výsledky praktických testů ukazují, že metoda BFGS vyžaduje menší počet operací než DFP. Proto je BFGS obecně považována za nejlepší známou newtonovskou metodu. Powell v roce 1976 také dokázal, že metoda BFGS při použití (2.15) pro hledání délky kroku je pro konvexní funkce globálně konvergentní (tj. končí vždy v lokálním minimu bez ohledu na volbu počátečního bodu).

Dá se ukázat, že počet iterací jak BFGS, tak i DFP nezávisí na *lineární transformaci proměnných.* Tedy pokud provedeme substituci $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{a}$, počet iterací potřebných k nalezení minima se nezmění, za předpokladu, že stejnou substituci provedeme i v použitém kritériu pro ukončení výpočtu. Z toho je zřejmé, že kvazi-newtonovské metody mohou fungovat dobře i v dlouhých úzkých údolích, která dělají problémy například metodě největšího spádu.

4 Metody sdružených gradientů

Metody sdružených gradientů jsou příkladem metod sdružených směrů, které mají podle věty 2.2 příznivé konvergenční vlastnosti přinejmenším pro kvadratické funkce. První krok metody sdružených gradientů je totožný s metodou největšího spádu, tj.

$$\mathbf{s}^{(1)} = -\mathbf{g}^{(1)}. \tag{4.1}$$

V k-té iteraci se nejdříve najde minimum funkce f ve směru $\mathbf{s}^{(k)}$:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{s}^{(k)}. \tag{4.2}$$

Pak se zvolí směr $\mathbf{s}^{(k+1)}$ tak, aby v případě, kdy f je kvadratická funkce s maticí $\mathbf{G}$ byl vektor $\mathbf{s}^{(k+1)}$ sdružený s $\mathbf{s}^{(1)}, \ldots \mathbf{s}^{(k)}$ a zároveň aby platila rovnice

$$\mathbf{s}^{(k+1)} = -\mathbf{g}^{(k+1)} + \beta^{(k)}\mathbf{s}^{(k)}, \tag{4.3}$$

kde $\beta^{(k)} \in \mathbb{R}$. Existuje více rozumných možností jak volit číslo $\beta^{(k)}$. V každém případě ale rovnice (4.3) vyjadřuje myšlenku, že $\mathbf{s}^{(k+1)}$ má být konvexní kombinací směru největšího spádu $(-\mathbf{g}^{(k)})$ a směru použitého v poslední iteraci. Pokud totiž položíme $\mu = 1/(1+\beta^{(k)})$, dostáváme

$$\mathbf{s}^{(k+1)} = \frac{1}{\mu}\left[\mu(-\mathbf{g}^{(k)}) + (1-\mu)\mathbf{s}^{(k)}\right].$$

Nyní odvodíme hodnotu $\beta^{(k)}$ za předpokladu, že minimalizovaná funkce je kvadratická, s kladně definitní Hessovou maticí $\mathbf{G}$. Podle (1.16) je

$$\mathbf{y}^{(k)} = \mathbf{G}\boldsymbol{\delta}^{(k)}, \tag{4.4}$$

kde $\mathbf{y}^{(k)} = \mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}$ a $\boldsymbol{\delta}^{(k)} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$. Protože vektory $\mathbf{s}^{(i)}$ mají být sdružené vzhledem k $\mathbf{G}$, musí platit ${\mathbf{s}^{(k+1)}}^{\mathrm{T}}\mathbf{G}\mathbf{s}^{(k)} = 0$. Podle (4.2) je $\boldsymbol{\delta}^{(k)}$ skalárním násobkem $\mathbf{s}^{(k)}$ a proto

$$0 = {\mathbf{s}^{(k+1)}}^{\mathrm{T}}\mathbf{G}\boldsymbol{\delta}^{(k)} \tag{4.5}$$

$$= {\mathbf{s}^{(k+1)}}^{\mathrm{T}}\mathbf{y}^{(k)} \quad \text{podle (4.4)}. \tag{4.6}$$

Po transponování rovnice (4.3) a násobení vektorem $\mathbf{y}^{(k)}$ zprava dostáváme

$$0 = -{\mathbf{g}^{(k+1)}}^{\mathrm{T}}\mathbf{y}^{(k)} + \beta^{(k)}{\mathbf{s}^{(k)}}^{\mathrm{T}}\mathbf{y}^{(k)}.$$

Odtud plyne hodnota $\beta^{(k)}$, kterou navrhli Hestenes a Stiefel v roce 1952:

$$\beta^{(k)}_{\mathrm{HS}} = \frac{{\mathbf{g}^{(k+1)}}^{\mathrm{T}}(\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)})}{{\mathbf{s}^{(k)}}^{\mathrm{T}}(\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)})}. \tag{4.7}$$

Pokud je $\mathbf{x}^{(k)}$ v (4.2) přesně určeným minimem ve směru $\mathbf{s}^{(k)}$, musí platit ${\mathbf{s}^{(k)}}^{\mathrm{T}}\mathbf{g}^{(k+1)} = 0$ (jinak by se hodnota f ve směru $\mathbf{s}^{(k)}$ dala ještě snížit), obdobně ${\mathbf{s}^{(k-1)}}^{\mathrm{T}}\mathbf{g}^{(k)} = 0$. Pak jmenovatel v (4.7) je roven

$${\mathbf{s}^{(k)}}^{\mathrm{T}}(\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}) = -{\mathbf{s}^{(k)}}^{\mathrm{T}}\mathbf{g}^{(k)} = \left[\mathbf{g}^{(k)} - \beta^{(k-1)}\mathbf{s}^{(k-1)}\right]^{\mathrm{T}}\mathbf{g}^{(k)} = \|\mathbf{g}^{(k)}\|_2^2.$$

Po dosazení do (4.7) vychází vyjádření $\beta^{(k)}$, které poprvé publikovali Polak a Ribiere v roce 1969:

$$\beta_{\mathrm{PR}}^{(k)} = \frac{\mathbf{g}^{(k+1)^{\mathrm{T}}}(\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)})}{\|\mathbf{g}^{(k)}\|_2^2}\,. \tag{4.8}$$

Obdobně se ukáže, že $\mathbf{g}^{(k+1)^{\mathrm{T}}}\mathbf{g}^{(k)} = 0$, odkud plyne $\beta^{(k)}$ podle Fletchera a Reevese (1963):

$$\beta_{\mathrm{FR}}^{(k)} = \frac{\|\mathbf{g}^{(k+1)}\|_2^2}{\|\mathbf{g}^{(k)}\|_2^2}\,. \tag{4.9}$$

Pro kvadratické funkce je $\beta_{\mathrm{HS}}^{(k)} = \beta_{\mathrm{PR}}^{(k)} = \beta_{\mathrm{FR}}^{(k)}$. To ale samozřejmě neplatí pro obecnější funkce. Při testovacích úlohách dává obvykle nejlepší výsledky varianta Polaka a Ribiere.

Zřejmou výhodou metod sdružených gradientů oproti newtonovským metodám jsou nižší paměťové nároky: stačí ukládat několik n-prvkových vektorů. Proto tyto metody dlouho představovaly jedinou možnost pro rozsáhlé optimalizační výpočty (například úlohy vyskytující se v počítačové chemii mohou mít až několik desítek tisíc proměnných). Teprve nedávno se objevily modifikace newtonovských metod, které nevyžadují ukládání hessiánu (aproximují jej z jisté posloupnosti vektorů). Také oblast aplikovatelnosti newtonovských metod s kvadratickou pamětí se rozšířila díky růstu kapacity počítačů.

Pro variantu Fletchera a Reevese jsou dokázány výsledky, které zaručují superlineární konvergenci v blízkosti řešení za podmínek obdobných jako u konvergenčních výsledků pro kvazi-newtonovské metody.

5 Metody s omezeným krokem

Základní myšlenkou newtonovských metod je hledání minima lokální kvadratické aproximace $q^{(k)}$ vyjádřené rovnicí (3.1). To v sobě ale skrývá některé problémy. Například hessián příslušné kvadratické funkce nemusí být kladně definitní a tedy kvadratická aproximace nemusí mít jednoznačně určené minimum. Tento nedostatek je odstraněn u kvazi-newtonovských metod, ale jen v případě, že nemáme možnost vypočítat přesnou hodnotu hessiánu minimalizované funkce (pokud *máme* efektivní proceduru pro výpočet hessiánu, máme ji zahodit jen proto, že v některých bodech nemusí být výsledek kladně definitní?). Funkce $q^{(k)}$ také prostě nemusí v okolí svého minima aproximovat dostatečně přesně minimalizovanou funkci.

Metody s omezeným krokem se snaží obejít uvedená omezení tím, že hledají krok metody (tj. vektor $\boldsymbol{\delta}^{(k)} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$) jako minimum funkce $q^{(k)}(\boldsymbol{\delta})$ *na omezené oblasti* o které se dá předpokládat, že v ní $q^{(k)}$ dostatečně přesně modeluje minimalizovanou funkci. Tato oblast má tvar n-rozměrné koule (vzhledem ke zvolené normě), se středem v bodě vypočteném v poslední iteraci:

$$\Omega^{(k)} = \{\mathbf{x};\ \|\mathbf{x} - \mathbf{x}^{(k)}\| \leq h^{(k)}\}. \tag{5.1}$$

V každé iteraci se pak řeší problém

$$\text{minimalizuj } q^{(k)}(\boldsymbol{\delta}) \quad \text{za podmínky} \quad \|\boldsymbol{\delta}\| \leq h^{(k)}. \tag{5.2}$$

$\Omega^{(k)}$ je tedy okolí bodu $\mathbf{x}^{(k)}$, ve kterém „věříme“ kvadratické aproximaci. Tomu odpovídá i jeden ze dvou anglických termínů pro metody s omezeným krokem, *trust region methods* (druhým je *restricted step methods*).

Hodnota $h^{(k)}$ je upravována v každé iteraci tak, aby byla co největší při zachování dostatečné shody mezi f a $q^{(k)}$. Velikost $h^{(k+1)}$ se oproti $h^{(k)}$ zmenšuje/zvětšuje na základě *skutečného poklesu* funkce f

$$\Delta f^{(k)} = f^{(k)} - f(\mathbf{x}^{(k)} + \boldsymbol{\delta}^{(k)}) \tag{5.3}$$

a na základě *předpovězeného poklesu*

$$\Delta q^{(k)} = q^{(k)}(\mathbf{0}) - q^{(k)}(\boldsymbol{\delta}^{(k)}) = f^{(k)} - q^{(k)}(\boldsymbol{\delta}^{(k)}). \tag{5.4}$$

Podíl

$$r^{(k)} = \frac{\Delta f^{(k)}}{\Delta q^{(k)}} \tag{5.5}$$

pak slouží pro odhad přesnosti s jakou $q^{(k)}(\boldsymbol{\delta}^{(k)})$ aproximuje $f(\mathbf{x}^{(k)} + \boldsymbol{\delta}^{(k)})$: čím je $r^{(k)}$ bližší k jedničce, tím se aproximace považuje za přesnější.

Modelový algoritmus pro metodu s omezeným krokem závisí na jistých pevně zvolených konstantách $0 < \alpha_1 < \alpha_2 < 1$ a $0 < \eta_1 < 1 < \eta_2$, které slouží k adaptivní úpravě maximální délky kroku.

$$
\begin{array}{ll}
\text{(a)} & \text{Z } \mathbf{x}^{(k)} \text{ vypočti } \mathbf{g}^{(k)} \text{ a } \mathbf{G}^{(k)}. \\
\text{(b)} & \text{Najdi } \boldsymbol{\delta}^{(k)} \text{ minimalizující } q^{(k)}(\boldsymbol{\delta}) \text{ při omezení } \|\boldsymbol{\delta}\| \leq h^{(k)}. \\
\text{(c)} & \text{Vypočti } f(\mathbf{x}^{(k)} + \boldsymbol{\delta}^{(k)}) \text{ a odtud } r^{(k)}. \\
\text{(d)} & \text{Pokud } r^{(k)} < \alpha_1, \text{ polož } h^{(k+1)} = \eta_1 h^{(k)}, \\
& \text{pokud } r^{(k)} > \alpha_2 \text{ a } \|\boldsymbol{\delta}^{(k)}\| = h^{(k)}, \text{ polož } h^{(k+1)} = \eta_2 h^{(k)}, \\
& \text{jinak polož } h^{(k+1)} = h^{(k)}. \\
\text{(e)} & \text{Pokud } r^{(k)} \leq 0, \text{ polož } \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}, \text{ jinak } \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}^{(k)}.
\end{array}
\tag{5.6}
$$

Konstanty $\alpha_1, \alpha_2, \eta_1, \eta_2$ lze volit značně volně, aniž by to podstatně ovlivnilo konvergenci algoritmu. Typickou hodnotou pro α_2 je 0.75. Dále můžeme volit například $\alpha_1 = 0.25$, $\eta_1 = 0.25$, $\eta_2 = 2$. Někteří autoři ale doporučují volit α_1 velmi malé, například $\alpha_1 = 10^{-4}$.

Věta 5.1. *Pokud pro posloupnost $\{\mathbf{x}^{(k)}\}$ definovanou algoritmem* (5.6) *platí $\mathbf{x}^{(k)} \in B \subset \mathbb{R}^n$ $\forall k$, kde B je omezená množina a pokud f má spojité druhé derivace na B, pak existuje hromadný bod $\mathbf{x}^{\infty}$ posloupnosti $\{\mathbf{x}^{(k)}\}$, který splňuje nutné podmínky pro lokální minimum funkce* (2.1) *a* (2.2).

Jestliže v bodě $\mathbf{x}^{\infty}$ navíc platí předpoklady věty 3.1, pak $\mathbf{x}^{\infty}$ je lokálním minimem, $\mathbf{x}^{(k)} \to \mathbf{x}^{\infty}$, $\inf h^{(k)} > 0$ a existuje k_0 tak, že pro $k > k_0$ se nevyužívá omezení $\|\boldsymbol{\delta}\| \leq h^{(k)}$ a $\{\mathbf{x}^{(k)}\}$ konverguje kvadraticky.

Uvedená věta působí dojmem velmi silného tvrzení: za rozumných předpokladů jsou metody s omezeným krokem globálně konvergentní a v okolí minima konvergují kvadraticky, protože přecházejí v Newtonovu metodu (která nemá omezení $\|\boldsymbol{\delta}\| \leq h^{(k)}$). Metody s omezeným krokem jsou skutečně také úspěšné v praxi. Nicméně tyto metody vyžadují v každé iteraci globální minimalizaci $q^{(k)}$ na jisté omezené oblasti, což vyžaduje nezanedbatelný objem výpočtů.

5.1 Metody Levenberga a Marquardta

Nejvíce jsou metody s omezeným krokem propracovány pro případ euklidovské (L_2) normy. Ukážeme, že se pak řešení omezeného minimalizačního problému pro $q^{(k)}$ dá určit pomocí řešení lineární soustavy

$$(\mathbf{G}^{(k)} + \nu\mathbf{I})\boldsymbol{\delta}^{(k)} = -\mathbf{g}^{(k)} \tag{5.7}$$

pro vhodné ν. Tento typ metod byl navržen Levenbergem (1944) a Marquardtem (1963) a sloužil zpočátku jen k řešení úlohy nejmenších čtverců (té se týká kapitola 6). Pro euklidovskou normu se dá (5.2) přepsat do tvaru

$$\begin{aligned} &\text{minimalizuj } q^{(k)}(\boldsymbol{\delta}) \stackrel{\text{def}}{=} \tfrac{1}{2}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{G}^{(k)}\boldsymbol{\delta} + {\mathbf{g}^{(k)}}^{\mathrm{T}}\boldsymbol{\delta} \\ &\text{za předpokladu } \boldsymbol{\delta}^{\mathrm{T}}\boldsymbol{\delta} \leq {h^{(k)}}^2. \end{aligned} \tag{5.8}$$

Věta 5.2. *(Moré a Sorensen 1982) Vektor $\boldsymbol{\delta}^{(k)}$ je globálním minimem problému* (5.8) *právě když existuje $\nu \geq 0$ tak, že platí* (5.7)*, je splněna rovnice*

$$\nu\left({h^{(k)}}^2 - {\boldsymbol{\delta}^{(k)}}^{\mathrm{T}}\boldsymbol{\delta}^{(k)}\right) = 0 \tag{5.9}$$

a matice $\mathbf{G}^{(k)} + \nu\mathbf{I}$ je kladně semidefinitní. Pokud navíc $\mathbf{G}^{(k)} + \nu\mathbf{I}$ je kladně definitní, je $\boldsymbol{\delta}^{(k)}$ jediným řešením (5.8)*.*

Všechny Levenberg-Marquardtovy algoritmy nejdříve najdou jisté $\nu \geq 0$ pro které je $\mathbf{G}^{(k)} + \nu\mathbf{I}$ kladně definitní (pro dostatečně velké ν to platí) a poté řeší rovnici (5.7). Pokud $\nu > 0$, musí platit $\|\boldsymbol{\delta}^{(k)}\|_2 = h^{(k)}$. Proto se $\boldsymbol{\delta}^{(k)}$ obvykle určuje pomocí iterativního procesu, ve kterém se řeší rovnice

$$\frac{1}{\|(\mathbf{G}^{(k)} + \nu\mathbf{I})^{-1}\mathbf{g}^{(k)}\|_2} - \frac{1}{h^{(k)}} = 0 \tag{5.10}$$

vzhledem k ν.

5.2 Metoda psí nohy

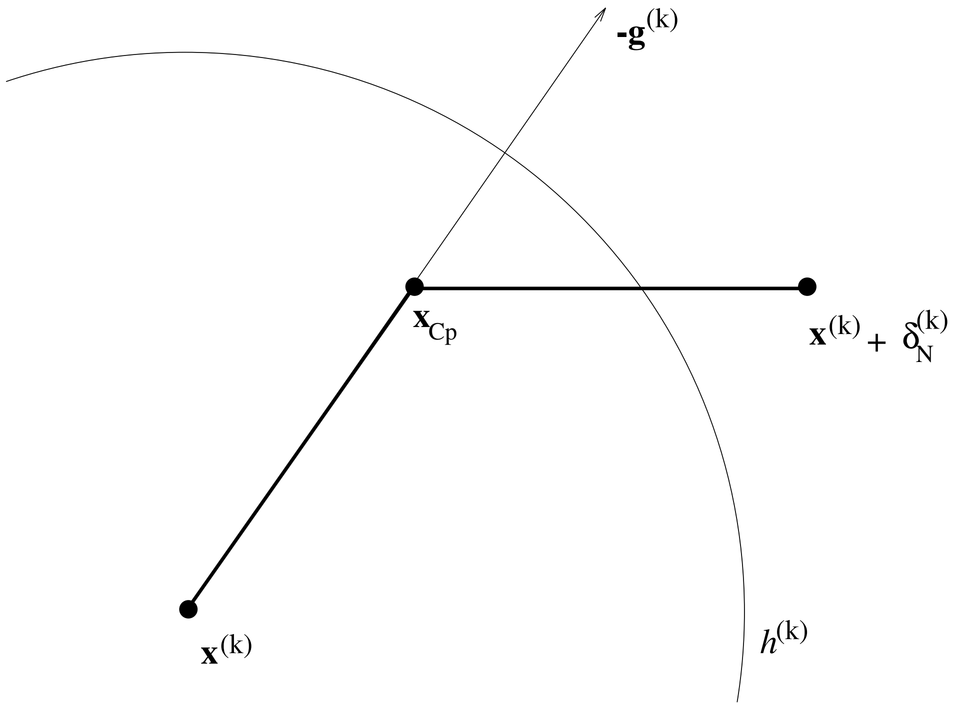
Je možné si všimnout, že pokud je krok $\boldsymbol{\delta}^{(k)}$ určován podle (5.7), pak pro $\nu = 0$ přechází výpočet v Newtonovu metodu, zatímco pro $\nu \to \infty$ definuje $\boldsymbol{\delta}^{(k)}$ směr největšího spádu. Metoda založená na uvedeném zjištění proto počítá svůj krok buď jako Newtonova metoda nebo jako metoda největšího spádu nebo jako jistou kombinaci kroků těchto dvou metod.

Stejně jako u Levenberg-Marquardtových metod je v k-té iteraci definováno jisté číslo $h^{(k)}$, které určuje oblast, ve které má kvadratická aproximace minimalizované funkce rozumnou platnost. *Metoda psí nohy* nejdříve spočítá newtonovský krok $\boldsymbol{\delta}_N^{(k)} = -{\mathbf{G}^{(k)}}^{-1}\mathbf{g}^{(k)}$ (viz (3.1) a (3.2)) a bod $\mathbf{x}_{\mathrm{Cp}}$ (nazývaný někdy Cauchyho bod), ve kterém nastává minimum kvadratické aproximace ve směru největšího spádu. Dá se ukázat, že

$$\mathbf{x}_{\mathrm{Cp}} = \mathbf{x}^{(k)} - \frac{\mathbf{g}^{(k)}\|\mathbf{g}^{(k)}\|_2^2}{{\mathbf{g}^{(k)}}^{\mathrm{T}}\mathbf{G}^{(k)}\mathbf{g}^{(k)}}\,.$$

Krok metody pak leží na lomené čáře spojující $\mathbf{x}^{(k)}$, $\mathbf{x}_{\mathrm{Cp}}$ a $\mathbf{x}_N$ (viz obrázek) a který je ve vzdálenosti $h^{(k)}$ od $\mathbf{x}^{(k)}$. Parametr $h^{(k)}$ se mění opět adaptivně podle algoritmu (5.6), ve kterém je jen krok (b) nahrazen výpočtem $\boldsymbol{\delta}^{(k)}$ podle právě popsaného postupu.

Metoda psí nohy je globálně konvergentní, tj. pro funkce, které splňují předpoklady věty 3.1 konverguje vždy ke stacionárnímu bodu.



Obrázek 2: Metoda psí nohy.

5.3 Srovnání s metodami spádových směrů

Nedá se jednoznačně říci, že by metody spádových směrů (tj. ty, které hledají další bod neohraničenou minimalizací podle jisté zvolené přímky) byly lepší nebo horší než metody s omezeným krokem. Z hlediska teorie sice pro metody s omezeným krokem existují silnější výsledky (věta 5.1) než pro newtonovské metody, ale v praxi se rozsah použitelnosti obou typů metod v podstatě neliší. Záleží tedy spíše na propracovanosti konkrétní implementace než na formálním zařazení metody.

Při „přihrádkování" minimalizačních metod je dobré si uvědomit, že dělení není tak ostré jak možná na první pohled vypadá. Například kvazi-newtonovské metody jsme popisovali jako typické metody spádových směrů. Přitom jejich podstatou není ani tak hledání ve spádovém směru jako spíše efektivní (a kladně definitní) aproximace hessiánu. Takto vypočítaná aproximace se dá snadno použít i v algoritmu 5.6 namísto matice $\mathbf{G}^{(k)}$. Pokud je aproximace vypočtena například pomocí metody BFGS, dostáváme metodu BFGS, která určuje krok pomocí metody s omezeným krokem. Obdobně můžeme pro hledání kroku v metodě BFGS použít metodu psí nohy.

6 Minimalizace součtu čtverců

Minimalizace odchylky teoretického modelu od naměřených hodnot je častou úlohou všude, kde máme co do činění s analýzou nějakého přírodního nebo technického systému. Naměřená data si můžeme představit jako dvojice

$$(\mathbf{t}_i, y_i), \quad i = 1, \dots, m,$$

kde $\mathbf{t}_i \in \mathbb{R}^k$ je bod měření (například *čas* nebo *místo* měření, nebo obojí) a $y_i \in \mathbb{R}$ je hodnota naměřená v $\mathbf{t}_i$. Dále pak máme nějaký matematický model $M : \mathbb{R}^{k+n} \to \mathbb{R}$, který je závislý na n volných parametrech $x_1, x_2, \dots, x_n$ a pro který požadujeme, aby

$$M(\mathbf{t}_i, \mathbf{x}) \approx y_i, \quad i = 1, \dots, m.$$

Chceme tedy, aby hodnoty rozdílů

$$r_i(\mathbf{x}) = M(\mathbf{t}_i, \mathbf{x}) - y_i \tag{6.1}$$

byly v absolutní hodnotě co nejmenší. To se dá interpretovat jako minimalizace normy vektoru

$$\mathbf{r}(\mathbf{x}) = (r_1(\mathbf{x}), \dots, r_m(\mathbf{x}))^{\mathrm{T}}. \tag{6.2}$$

Nejčastěji se používá euklidovská (L_2) norma, pro kterou dostáváme minimalizovanou funkci ve tvaru

$$f(x) = \mathbf{r}(\mathbf{x})^{\mathrm{T}}\mathbf{r}(\mathbf{x}) = \sum_{i=1}^{m} [r_i(\mathbf{x})]^2. \tag{6.3}$$

Namísto L_2 normy je možné použít také normu L_1 (součet absolutních hodnot r_i), nebo L_∞ (maximum z absolutních hodnot r_i). Tyto normy *mají* svoje opodstatnění: například L_1 norma lépe eliminuje body měření, které „uletěly“, tj. jsou výrazně mimo průběh zadaný ostatními body, často v důsledku chyby při měření. V obou případech ale minimalizovaná funkce není všude diferencovatelná a proto příslušný minimalizační problém vyžaduje speciální výklad. Základní metody používané pro L_1 normu se dají najít ve Fletcherově učebnici. Zde se dále budeme zabývat pouze případem (6.3).

Na funkce typu (6.3) je možné přímo aplikovat minimalizační metody probrané v předchozích kapitolách. Při výpočtech ale můžeme ušetřit mnoho času i paměti tím, že využijeme speciální vlastnosti tohoto problému. Gradient funkce f se dá vyjádřit jako

$$\mathbf{g}(\mathbf{x}) = 2\sum_{i=1}^{m} \nabla r_i(\mathbf{x}) r_i(\mathbf{x}) = 2\,\mathbf{J}(\mathbf{x})^{\mathrm{T}}\mathbf{r}(\mathbf{x}), \tag{6.4}$$

kde $\mathbf{J}(x) \in \mathbb{R}^{m \times n}$ je *Jakobiho matice* funkce $\mathbf{r}(x)$ (i-tý řádek matice $\mathbf{J}(x)$ je gradient r_i v bodě x). Hessián funkce f se pak dá zapsat jako

$$\mathbf{G}(\mathbf{x}) = 2\sum_{i=1}^{m} \nabla r_i(\mathbf{x}) \nabla r_i(\mathbf{x})^{\mathrm{T}} + 2\sum_{i=1}^{m} \nabla^2 r_i(\mathbf{x}) r_i(\mathbf{x}). \tag{6.5}$$

Pokud je model m v dobré shodě s daty, pak v minimu $\mathbf{x}^*$ má funkce f velmi malou (kladnou) hodnotu. Pak tedy $r_i(\mathbf{x})$ jsou malá čísla a v okolí $\mathbf{x}^*$ se proto dá zanedbat druhá suma v (6.5). Hessián funkce f pak může být aproximován jako

$$\mathbf{G}(\mathbf{x}) \approx 2 \sum_{i=1}^{m} \nabla r_i(\mathbf{x}) \nabla r_i(\mathbf{x})^{\mathrm{T}} = 2\, \mathbf{J}(\mathbf{x})^{\mathrm{T}} \mathbf{J}(\mathbf{x}). \tag{6.6}$$

Metody, které kombinují aproximaci (6.6) s hledáním ve spádovém směru se nazývají *Gauss-Newtonovy*. Spádový směr je pak určen řešením rovnice

$$\mathbf{J}^{(k)\mathrm{T}} \mathbf{J}^{(k)} \mathbf{s}^{(k)} = -\mathbf{J}^{(k)\mathrm{T}} \mathbf{r}^{(k)}. \tag{6.7}$$

Použití (6.6) v metodě s omezeným krokem dává *Levenberg-Marquardtovu metodu pro součet čtverců*. Původně byla Levenberg-Marquardtova metoda vyvinuta právě pro tuto aplikaci. Podle (5.7) se v tomto speciálním případě řeší rovnice

$$\left(\mathbf{J}^{(k)\mathrm{T}} \mathbf{J}^{(k)} + \nu \mathbf{I}\right) \boldsymbol{\delta}^{(k)} = -\mathbf{J}^{(k)\mathrm{T}} \mathbf{r}^{(k)}. \tag{6.8}$$

Situace se komplikuje v případě, kdy neexistuje dostatečně dobré minimum funkce f, tj. když $\mathbf{r}(\mathbf{x})$ je nezanedbatelně velké pro všechna $\mathbf{x}$. Pak je možné například kombinovat metodu BFGS s Gauss-Newtonovou metodou: podle velikosti relativního poklesu funkce f v k-té iteraci se hessián pro další iteraci aproximuje buď pomocí metody BFGS nebo pomocí Gauss-Newtonovy metody (Fletcher a Xu 1985). Přesněji řečeno, pokud

$$f^{(k)} - f^{(k+1)} \geq \tau f^{(k)}, \tag{6.9}$$

kde $\tau \in (0, 1)$ je vhodně zvolený parametr, pak se pro odhad použije Gauss-Newtonova metoda, v opačném případě pak BFGS. Fletcher (1987) doporučuje volit $\tau = 0.2$.

7 Nelineární optimalizace s omezeními

Většina spojitých optimalizačních problémů se dá vyjádřit následujícím způsobem:

$$\begin{array}{lll} \text{minimalizuj} & f(\mathbf{x}) & \mathbf{x} \in \mathbb{R}^n \\ \text{za předpokladu} & c_i(\mathbf{x}) = 0, & i \in E \\ & c_i(\mathbf{x}) \geq 0, & i \in I. \end{array} \tag{7.1}$$

Účelová funkce f, stejně jako funkce c_i určující omezení jsou zobrazení z $\mathbb{R}^n$ do $\mathbb{R}$. Budeme předpokládat, že f a c_i jsou spojitě diferencovatelné. V dalším textu budeme dodržovat označení indexových množin z (7.1), tj. E bude vždy označovat množinu omezení ve formě rovnic a I soustavu omezení ve formě nerovnic. Dále budeme kvůli stručnosti označovat gradienty ∇c_i jako $\mathbf{a}_i$.

Bod, který splňuje omezující podmínky (7.1) nazveme *přípustným bodem.* Přípustný bod $\mathbf{x}^*$ splňující omezující podmínky nazveme *omezeným lokálním minimem* problému (7.1), pokud pro všechny přípustné body $\mathbf{x}$ dostatečně blízké k $\mathbf{x}^*$ platí $f(\mathbf{x}^*) \leq f(\mathbf{x})$.

Důležitým pojmem jsou *aktivní omezení.* Aktivní omezení v daném bodě $\mathbf{x}$ jsou definována jako indexová množina

$$\mathcal{A}(\mathbf{x}) = \{i;\, c_i(\mathbf{x}) = 0\}. \tag{7.2}$$

Je-li tedy $i \in \mathcal{A}(\mathbf{x})$, pak $\mathbf{x}$ je na hranici oblasti definované i-tým omezením. Zvlášť důležitá je množina aktivních omezení v řešení $\mathbf{x}^*$ úlohy (7.1), kterou budeme označovat $\mathcal{A}^*$. Jestliže je tato množina známa, pak můžeme ostatní omezení ignorovat (v jistém okolí bodu $\mathbf{x}^*$) a řešit úlohu jako problém s omezeními ve formě rovnic, $E = \mathcal{A}^*$.

7.1 Eliminace proměnných

Jestliže jsou omezení tvořena pouze soustavou m rovnic, můžeme z nich vyjádřit m proměnných pomocí zbývajících $n-m$ proměnných (jsou-li splněny předpoklady věty o implicitní funkci). Za těchto m proměnných pak můžeme dosadit v účelové funkci a řešit namísto původního problému o n proměnných s omezeními nepodmíněný minimalizační problém o $n-m$ proměnných. Pokud řešení nepodmíněného problému splňuje omezení původní úlohy, dospěli jsme k řešení. Takto vytvořený nepodmíněný problém ale obecně *nemusí* být ekvivalentní původnímu problému a proto je třeba při eliminaci proměnných zachovávat jistou opatrnost a konfrontovat nalezené řešení s původním zadáním.

Příkladem minimalizace, kdy uvedený přístup funguje, je $\min -x - y$ za předpokladu $x^2 + y^2 = 1$. Opačným příkladem je minimalizace $x^2 + y^2$ za předpokladu $(x-1)^3 = y^2$ (v tomto případě je výsledná nepodmíněná účelová funkce zdola neomezená). Vyzkoušejte řešení obou problémů jak graficky tak pomocí eliminace proměnných a zjistěte, proč v druhém případě eliminace proměnných selhává.

7.2 Omezení v podobě rovnic

Ukážeme odtud, že pokud máme pouze omexení ve tvaru rovnic ($I = \emptyset$) a vektory $\mathbf{a}_i^*$ jsou lineárně nezávislé, musí pro každé omezené lokální minimum $\mathbf{x}^*$ existovat čísla λ_i^*, $i \in E$ taková, že

$$\mathbf{g}^* = \sum_{i \in E} \mathbf{a}_i^* \lambda_i^*. \tag{7.3}$$

Věta 7.1. *(Věta o implicitní funkci) Buď $F = (f_1, \dots, f_m) : G \subset \mathbb{R}^{m+k} \to \mathbb{R}^m$ funkce, která má na otevřené množině G spojité první derivace. Pokud v bodě $\mathbf{x}^0 \in G$ platí $F(\mathbf{x}^0) = \mathbf{0}$ a jakobián*

$$\begin{vmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_m} \\ \dots & \dots & \dots \\ \frac{\partial f_m}{x_1} & \dots & \frac{\partial f_m}{x_m} \end{vmatrix}$$

je různý od nuly v $\mathbf{x}^0$, pak v jistém okolí Ω bodu $(x^0_{m+1}, \dots, x^0_{m+k})$ existují spojité reálné funkce $\varphi = (\varphi_1, \dots, \varphi_m)$ takové, že $F(\varphi(\mathbf{z}), \mathbf{z}) = \mathbf{0}$ pro každé $\mathbf{z} \in \Omega$. Navíc pokud $F(\mathbf{y}, \mathbf{z}) = \mathbf{0}$, kde $\mathbf{z} \in \Omega$, pak $\mathbf{y} = \varphi(\mathbf{z})$.

Věta 7.2. *Mějme problém minimalizace funkce $f(x)$ za předpokladu $c_i(x) = 0$ $(i = 1, \dots, m)$, kde $f : \mathbb{R}^n \to \mathbb{R}$ a $g_i : \mathbb{R}^n \to \mathbb{R}$ jsou funkce se spojitými prvními derivacemi. Je-li pak $\mathbf{x}^*$ omezené lokální minimum tohoto problému a gradienty $\mathbf{a}^*_i = \nabla c_i(\mathbf{x}^*)$ jsou lineárně nezávislé, pak $\nabla f(\mathbf{x}^*)$ je lineární kombinací vektorů $\mathbf{a}^*_1, \dots, \mathbf{a}^*_m$.*

Důkaz. Protože vektory $\mathbf{a}^*_i$ $(i = 1, \dots, m)$ jsou lineárně nezávislé, lze z Jakobiho matice

$$\mathbf{J} = \begin{pmatrix} {\mathbf{a}^*_1}^{\mathrm{T}} \\ \dots \\ {\mathbf{a}^*_m}^{\mathrm{T}} \end{pmatrix}$$

vybrat m lineárně nezávislých sloupců. Při vhodném očíslování proměnných to bude prvních m sloupců. Čtvercová matice

$$\begin{bmatrix} a^*_{1,1} & \cdots & a^*_{1,m} \\ \dots & \dots & \dots \\ a^*_{m,1} & \cdots & a^*_{m,m} \end{bmatrix}$$

je pak regulární a její determinant je tedy různý od nuly. Podle věty o implicitní funkci pak v jistém okolí Ω bodu $(x^*_{m+1}, \dots, x^*_n)$ existují spojité funkce φ_i $(i = 1, \dots, m)$ takové, že položíme-li $x_i = \varphi(x_{m+1}, \dots, x_n)$ $(i = 1, \dots, m)$, je $(x_1, \dots, x_n)$ řešením soustavy $c_i(x) = 0$ $(i = 1, \dots, m)$, tj. přípustným bodem úlohy. Navíc $(\varphi(\mathbf{z}), \mathbf{z})$ je jediný přípustný bod pro dané $\mathbf{z} \in \Omega$. V jistém okolí bodu $\mathbf{x}^*$ můžeme proto původní úlohu s omezeními nahradit minimalizací funkce

$$h(x_{m+1}, \dots, x_n) = f(\varphi_1(x_{m+1}, \dots, x_n), \dots, \varphi_m(x_{m+1}, \dots, x_n), x_{m+1}, \dots, x_n)$$

bez omezení.

Protože vektory $(a^*_{1,1}, \dots, a^*_{1,m}), \dots, (a^*_{1,1}, \dots, a^*_{1,m})$ jsou lineárně nezávislé, existují koeficienty $\lambda_1, \dots, \lambda_m$ takové, že

$$\frac{\partial f(\mathbf{x}^*)}{\partial x_j} = \sum_{i=1}^{m} \lambda_i a^*_{i,j} \quad (j = 1, \dots, m). \tag{7.4}$$

Využitím pravidla o derivování složené funkce se pak dá odvodit, že rovnice (7.4) platí i pro $j = m + 1, \dots, n$, tj. že

$$\nabla f(\mathbf{x}^*) = \sum_{i=1}^{m} \lambda_i \mathbf{a}^*_i \,.$$

Podrobnosti důkazu lze najít v Jarníkově učebnici *Diferenciální počet II* (3.vyd. Academia 1984) na straně 515.

□

Z podmínky (7.3) vychází klasická *metoda Lagrangeových činitelů* určená pro řešení úloh s omezeními v podobě rovnic. Cílem této metody je najít vektory $\mathbf{x}^*$ a $\boldsymbol{\lambda}^*$ splňující rovnice

$$\begin{aligned} \mathbf{g}(\mathbf{x}) &= \sum_{i \in E} \mathbf{a}_i(\mathbf{x}) \lambda_i, \\ c_i(\mathbf{x}) &= 0, \quad i \in E. \end{aligned} \tag{7.5}$$

Pokud máme m omezení ($|E| = m$), pak dostáváme $n+m$ rovnic o $n+m$ neznámých. Nicméně jde o systém nelineárních rovnic, takže nalezení řešení obecně nemusí být jednoduché. Při odvozování (7.5) jsme navíc předpokládali jen že $\mathbf{x}^*$ je stacionárním bodem, tj. $\mathbf{g}(\mathbf{x}^*) = \mathbf{0}$. Řešením (7.5) tedy mohou být kromě lokálních minim také lokální maxima a sedlové body.

Použití metody Lagrangeových činitelů si můžeme ukázat na dvoudimenzionálním příkladě. Máme minimalizovat $x_1 + x_2$ za předpokladu $x_1^2 - x_2 = 0$. Podle (7.5) sestavíme tři rovnice:

$$\begin{gathered} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2x_1 \\ -1 \end{pmatrix} \lambda \\ x_1^2 - x_2 = 0. \end{gathered}$$

Odtud dostáváme $\lambda^* = -1$, $x_1^* = -\frac{1}{2}$ a $x_2^* = \frac{1}{4}$.

Často je výhodné vyjádřit podmínku (7.5) pomocí *Lagrangeovy funkce*, neboli *lagrangeiánu*, definovaného jako

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) - \sum_i \lambda_i c_i(\mathbf{x}). \tag{7.6}$$

Pak je možné napsat (7.5) ve tvaru

$$\nabla \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \mathbf{0}, \tag{7.7}$$

kde

$$\nabla = \begin{pmatrix} \nabla_{\mathbf{x}} \\ \nabla_{\boldsymbol{\lambda}} \end{pmatrix}.$$

Je důležité vědět, že hodnoty Lagrangeových činitelů určují *citlivost* řešení vzhledem k perturbacím, tj. vzhledem ke změnám pravých stran v (7.1). Uvažujme modifikovaný problém, ve kterém na pravých stranách rovnic (opět budeme předpokládat, že náš problém neobsahuje nerovnice) jsou místo nul čísla ε_i, tj. máme soustavu

$$c_i(\mathbf{x}) = \varepsilon_i, \quad i \in E. \tag{7.8}$$

Bude nás nyní zajímat, jak se mění řešení $\mathbf{x}^*$ a $\boldsymbol{\lambda}^*$ v závislosti na vektoru $\boldsymbol{\varepsilon}$. Langrangeián pro modifikovaný problém je dán jako

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\varepsilon}) = f(\mathbf{x}) - \sum_i \lambda_i (c_i(\mathbf{x}) - \varepsilon_i). \tag{7.9}$$

Nechť $\mathbf{x}^* = \mathbf{x}^*(\boldsymbol{\varepsilon})$ je řešení problému $\min f(\mathbf{x})$ s omezeními (7.8) a $\boldsymbol{\lambda}^* = \boldsymbol{\lambda}(\boldsymbol{\varepsilon})$ je spolu s $\mathbf{x}^*$ řešením (7.7) (s výjimkou singulárních případů je $\boldsymbol{\lambda}^*$ určeno pro dané $\mathbf{x}^*$ jednoznačně). Protože $\mathbf{x}^*$ je přípustým bodem (7.8), platí $f(\mathbf{x}^*(\boldsymbol{\varepsilon})) = \mathcal{L}(\mathbf{x}^*(\boldsymbol{\varepsilon}), \boldsymbol{\lambda}^*(\boldsymbol{\varepsilon}), \boldsymbol{\varepsilon})$ a tedy

$$\frac{df(\mathbf{x}^*(\boldsymbol{\varepsilon}))}{d\varepsilon_i} = \frac{d\mathcal{L}(\mathbf{x}^*(\boldsymbol{\varepsilon}), \boldsymbol{\lambda}^*(\boldsymbol{\varepsilon}), \boldsymbol{\varepsilon})}{d\varepsilon_i}.$$

Výraz $\mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\varepsilon})$ můžeme chápat jako složenou funkci, $\mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\varepsilon}) = \mathcal{L}(\Psi(\boldsymbol{\varepsilon}))$, kde $\Psi(\boldsymbol{\varepsilon}) = (\mathbf{x}^*(\boldsymbol{\varepsilon}), \boldsymbol{\lambda}^*(\boldsymbol{\varepsilon}), \boldsymbol{\varepsilon})$. Proto

$$\frac{d\mathcal{L}}{d\varepsilon_i} = \left[\frac{\partial \Psi}{\partial \varepsilon_i}\right]^{\mathrm{T}} \nabla_{\Psi} \mathcal{L} = \left(\frac{\partial \mathbf{x}^*(\boldsymbol{\varepsilon})^{\mathrm{T}}}{\partial \varepsilon_i}, \frac{\partial \boldsymbol{\lambda}^*(\boldsymbol{\varepsilon})^{\mathrm{T}}}{\partial \varepsilon_i}, \frac{\partial \boldsymbol{\varepsilon}^{\mathrm{T}}}{\partial \varepsilon_i}\right) \begin{pmatrix} \nabla_{\mathbf{x}^*} \mathcal{L} \\ \nabla_{\boldsymbol{\lambda}^*} \mathcal{L} \\ \nabla_{\boldsymbol{\varepsilon}} \mathcal{L} \end{pmatrix}.$$

Protože podle předpokladu je $\nabla_{\mathbf{x}^*} \mathcal{L} = \nabla_{\boldsymbol{\lambda}^*} \mathcal{L} = \mathbf{0}$, platí

$$\frac{df}{d\varepsilon_i} = \frac{d\mathcal{L}}{d\varepsilon_i} = \frac{\partial \boldsymbol{\varepsilon}^{\mathrm{T}}}{\partial \varepsilon_i} \nabla_{\boldsymbol{\varepsilon}} \mathcal{L} = \mathbf{e}_i^{\mathrm{T}} \boldsymbol{\lambda} = \lambda_i \tag{7.10}$$

(kde $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*(\boldsymbol{\varepsilon})$). Pokud tedy budeme volit $\boldsymbol{\varepsilon}$ v jistém okolí bodu $\mathbf{0}$, můžeme odhadnout změnu hodnoty účelové funkce v řešení $\mathbf{x}^*(\boldsymbol{\varepsilon})$ oproti bodu $\mathbf{x}^* = \mathbf{x}^*(\mathbf{0})$ jako $\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{\varepsilon}$, kde $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*(\mathbf{0})$ jsou Lagrangeovy činitele pro původní nemodifikovanou úlohu.

7.3 Omezení v podobě nerovnic

Nyní se rozšíříme naše úvahy z předchozího odstavce i na případ, kdy soustava omezení obsahuje nerovnice. Nechť opět $\mathbf{x}^*$ je řešení úlohy (7.1). V okolí bodu $\mathbf{x}^*$ jsou důležité jen ty nerovnice, které představují aktivní omezení v $\mathbf{x}^*$. Označíme je jako I^* $(= \mathcal{A}^* \cap I)$.

Ukážeme, že pokud $\mathbf{x}^*$ je lokálním minimem (7.1), pak musí splňovat následující podmínky:

$$\mathbf{g}^* = \sum_{i \in \mathcal{A}^*} \mathbf{a}_i^* \lambda_i^* \tag{7.11}$$

a zároveň

$$\lambda_i^* \geq 0, \quad i \in I^*. \tag{7.12}$$

Toto tvrzení budeme dokazovat na základě výsledků z předchozího odstavce a budeme tedy předpokládat, že vektory $\mathbf{a}_i$ jsou v $\mathbf{x}^*$ (a tedy i v jistém jeho okolí) lineárně nezávislé.

Je-li $\mathbf{x}^*$ lokální minimum problému (7.1), pak z (7.3) plyne, že musí platit (7.11), protože aktivní omezení $c_i(\mathbf{x}) \geq 0$ přecházejí v bodě $\mathbf{x}^*$ v rovnice $c_i(\mathbf{x}) = 0$. Zbývá tedy dokázat, že Langrangeovy činitele v (7.11) jsou nezáporné. Představme si opět, že původní úlohu perturbujeme, a to tak, že pravou stranu jedné z rovnic $c_i(\mathbf{x}) = 0$ $(i \in \mathcal{A}^*)$ nahradíme číslem $\varepsilon_i > 0$. Pokud bude ε_i dostatečně malé, bude existovat řešení modifikované úlohy $\mathbf{x}^*(\varepsilon_i)$, které má stejnou množinu aktivních omezení, tj. $c_i(\mathbf{x}^*(\varepsilon_i)) = \varepsilon_i$ a $I^*(\varepsilon_i) = I^*$. Protože $\mathbf{x}^*(\varepsilon_i)$ je zároveň přípustným řešením původní úlohy, musí být $f(\mathbf{x}^*(\varepsilon_i)) \geq f(\mathbf{x}^*(0)) = f(\mathbf{x}^*)$. Pro $\lim_{\varepsilon_i \to 0}$ dostáváme

$$\frac{df(\mathbf{x}^*(\varepsilon_i))}{d\varepsilon_i} \geq 0$$

a tedy podle (7.10) je $\lambda_i^* \geq 0$.

Použití podmínek (7.11) a (7.12) si můžeme ukázat na příkladu:

$$\begin{aligned} &\text{minimalizuj} && f(\mathbf{x}) \stackrel{\text{def}}{=} -x_1 - x_2 \\ &\text{za předpokladu} && c_1(\mathbf{x}) \stackrel{\text{def}}{=} x_2 - x_1^2 \geq 0 \\ & && c_2(\mathbf{x}) \stackrel{\text{def}}{=} 1 - x_1^2 - x_2^2 \geq 0 \end{aligned} \tag{7.13}$$

Nemáme žádný obecný návod, jak zjistit, které z omezení je aktivní v $\mathbf{x}^*$. V daném případě se to dá poznat například tak, že si nakreslíme obrázek. Z něj snadno vidíme, že v řešení je aktivní pouze omezení 2. Protože $\nabla f = (-1, -1)^{\mathrm{T}}$, $\mathbf{x}^*$ musí být řešením soustavy

$$\begin{aligned} -1 &= -2\lambda_2 x_1 \\ -1 &= -2\lambda_2 x_2 \\ 1 - x_1^2 - x_2^2 &= 0 \end{aligned}$$

Tato soustava má dvě řešení, ale jen u jednoho z nich je $\lambda_2 \geq 0$:

$$\lambda_2 = x_1 = x_2 = \frac{1}{\sqrt{2}}.$$

Můžeme také postupovat „systematicky“ a zkoušet postupně všechny možnosti, které připadají v úvahu pro $\mathcal{A}^*$. Snadno lze vyloučit $\mathcal{A}^* = \emptyset$. Pro $\mathcal{A}^* = \{1\}$ neexistuje řešení, pro které by bylo $\lambda_1 \geq 0$. Pro $\mathcal{A}^* = \{1, 2\}$ neexistuje řešení, ve kterém by bylo *zároveň* $\lambda_1 \geq 0$ a $\lambda_2 \geq 0$. Tedy zbývá jediná možnost, $\mathcal{A}^* = \{2\}$. Je vidět, že systematická analýza i takto jednoduchého příkladu může zabrat dost času, zvlášť pokud máme k dispozici jen tužku a papír.

7.4 Podmínky pro vázané extrémy

V předchozích dvou odstavcích jsme ukázali, jakým způsobem se dají odvodit nutné podmínky pro lokální minima problému (7.1). Dosažené výsledky nyní shrneme do jednoho tvrzení. Termín *podmínky regularity*, který se vyskytuje v jeho předpokladech, bude pro naše účely znamenat, že:

- vektory $\mathbf{a}_i^* = \nabla c_i(\mathbf{x}^*)$ jsou lineárně nezávislé, *nebo*
- všechna omezení jsou lineární.

Věta 7.3. *(Kuhn-Tuckerovy podmínky) Je-li $\mathbf{x}^*$ lokálním minimem problému* (7.1) *a pokud v $\mathbf{x}^*$ platí podmínky regularity, pak existují Lagrangeovy činitele $\boldsymbol{\lambda}^*$ tak, že $\mathbf{x}^*$ a $\boldsymbol{\lambda}^*$ splňují následující soustavu omezení:*

$$\begin{aligned} \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) &= \mathbf{0} \\ c_i(\mathbf{x}) &= 0, \quad i \in E \\ c_i(\mathbf{x}) &\geq 0, \quad i \in I \\ \lambda_i &\geq 0, \quad i \in I \\ \lambda_i c_i(\mathbf{x}) &= 0 \quad \forall i. \end{aligned} \tag{7.14}$$

Rovnici $\nabla_{\mathbf{x}}\mathscr{L}(\mathbf{x},\boldsymbol{\lambda}) = \mathbf{0}$ v (7.14) dostáváme z (7.11), s tím, že předpokládáme, že pro neaktivní omezení je $\lambda_i^* = 0$. Tento dodatečný předpoklad je vyjádřen v poslední podmínce, $\lambda_i^* c_i(\mathbf{x}^*) = 0$, která říká, že $c_i(\mathbf{x}^*)$ a λ_i^* nemohou být zároveň nenulové.

Věta 7.3 bývá obvykle uváděna se slabšími předpoklady, než jsou naše „podmínky regularity“. Tyto předpoklady mohou být vyjádřeny například jako podmínka

$$\mathcal{F}^* \cap \mathcal{D}^* = F^* \cap \mathcal{D}^*, \tag{7.15}$$

kde
$\mathcal{D}^*$ je množina klesajících směrů v bodě $\mathbf{x}^*$, $\mathcal{D}^* = \{\mathbf{s};\ \mathbf{s}^{\mathrm{T}}\mathbf{g}^* < 0\}$,
$\mathcal{F}^*$ je množina směrů, které jsou limitou nějaké posloupnosti $\{\mathbf{s}^{(k)}\}$ takové, že existují $\boldsymbol{\delta}^{(k)} > 0$:

$$\mathbf{x}^{(k)} - \mathbf{x}^* = \boldsymbol{\delta}^{(k)}\mathbf{s}^{(k)} \qquad \forall k,$$

$F^* = \{\mathbf{s};\ \mathbf{s} \neq \mathbf{0},\ \mathbf{s}^{\mathrm{T}}\mathbf{a}_i^* = 0\,(i \in E),\ \mathbf{s}^{\mathrm{T}}\mathbf{a}_i^* \geq 0\,(i \in I^*)\}$ je množina přípustných směrů pro úlohu s linearizovanými omezeními (funkce $c_i(\mathbf{x})$ nahradíme lineárními kombinacemi $\mathbf{a}_i^{*\mathrm{T}}\mathbf{x}$). Dá se ukázat, že z výše uvedených podmínek regularity plyne $\mathcal{F}^* = F^*$. Ověření, že vektory $\mathbf{a}_i^*$ jsou lineárně nezávislé je nejdůležitější metodou pro testování podmínky (7.15).

Věta 7.4. *(Dostatečné podmínky 2.řádu) Nechť pro $\mathbf{x}^*$ existuje λ^* tak, že platí Kuhn-Tuckerovy podmínky* (7.14) *a platí*

$$\mathbf{s}^{\mathrm{T}}\nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}^*,\lambda^*)\mathbf{s} > 0 \tag{7.16}$$

pro všechna $\mathbf{s} \neq \mathbf{0}$ taková, že

$$\begin{aligned} \mathbf{s}^{\mathrm{T}}\mathbf{a}_i^* &= 0 \qquad && \textit{pro } i \in E \cup I_+^*, \\ \mathbf{s}^{\mathrm{T}}\mathbf{a}_i^* &\geq 0 \qquad && \textit{pro } i \in I_0^*, \end{aligned}$$

kde

$$\begin{aligned} I_+^* &= \{i;\ i \in I^*,\ \lambda_i^* > 0\}, && \\ I_0^* &= \{i;\ i \in I^*,\ \lambda_i^* = 0\} \qquad && \textit{(zde } I^* = \mathcal{A}^* \cap I\textit{).} \end{aligned}$$

Pak $\mathbf{x}^*$ je ostré lokální minimum úlohy (7.1).

7.5 Konvexní programování

Konvexní programování se zabývá minimalizací konvexní funkce na konvexní množině. Za těchto předpokladů umíme sestavit rovnice, jejichž řešením je *globální* minimum (jsou-li všechny funkce vyskytující se ve formulaci problému spojitě diferencovatelné, ale to je v praxi obvykle splněno).

$K \subset \mathbb{R}^n$ je *konvexní množina*, pokud z $\mathbf{x}, \mathbf{y}$ plyne

$$(1-\lambda)\mathbf{x} + \lambda\mathbf{y} \in K \quad \forall\lambda \in [0,1].$$

$f : S \subset \mathbb{R}^n \to \mathbb{R}$ je *konvexní funkce*, pokud pro každé $\mathbf{x}, \mathbf{y} \in S$ platí

$$f(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1-\lambda)f(\mathbf{y}) \quad \forall\lambda \in [0,1].$$

Funkce je tedy konvexní právě když přímka spojující body $(\mathbf{x}, f(\mathbf{x}))$ a $(\mathbf{y}, f(\mathbf{y}))$ shora ohraničuje funkci f na úsečce spojující $\mathbf{x}$ a $\mathbf{y}$. To je také právě když je množina (v $\mathbb{R}^{n+1}$) ležící nad grafem funkce f (*epigraf* funkce f) konvexní. O funkci f řekneme, že je *konkávní*, pokud je funkce $-f$ konvexní.

Nyní můžeme přesně zformulovat *úlohu konvexního programování*:

$$\begin{aligned} &\text{minimalizuj} \quad f(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n \\ &\text{za předpokladu} \ \ c_i(\mathbf{x}) \geq 0, \quad i = 1, \ldots, m, \end{aligned} \tag{7.17}$$

kde f je konvexní a c_i jsou konkávní.

Věta 7.5. *Každé lokální řešení úlohy* (7.17) *je zároveň globálním řešením a množina všech řešení je konvexní.*

Věta 7.6. *Pokud jsou funkce f a c_i ($i = 1, \ldots, m$) spojitě diferencovatelné a pokud platí Kuhn-Tuckerovy podmínky* (7.14) *pro bod $\mathbf{x}^* \in \mathbb{R}^n$, pak $\mathbf{x}^*$ je globálním minimem problému konvexního programování* (7.17).

7.6 Kvadratické programování

Přirozeným zobecněním lineárního programování je *úloha kvdratického programování*:

$$\begin{aligned} &\text{minimalizuj} && q(\mathbf{x}) \stackrel{\text{def}}{=} \tfrac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{G}\mathbf{x} + \mathbf{g}^{\mathrm{T}}\mathbf{x} \\ &\text{za předpokladu} && a_i^{\mathrm{T}}\mathbf{x} = b_i, \quad i \in E \\ &&& a_i^{\mathrm{T}}\mathbf{x} \geq b_i, \quad i \in I. \end{aligned} \tag{7.18}$$

Tato úloha má význam nejen sama o sobě, ale také proto, že některé metody pro nelineární programování s omezeními převádějí obecný problém (7.1) na posloupnost problémů typu (7.18). Pokud je Hessova matice $\mathbf{G}$ kladně semidefinitní, pak je funkce q konvexní a (7.18) je speciálním případem úlohy konvexního programování a tedy má je jediné minimum. Naopak úloha

$$\min\{-\mathbf{x}^{\mathrm{T}}\mathbf{x};\ -1 \leq \mathbf{x}_i \leq 1,\ i = 1, \ldots, n\}$$

má lokální minimum ve všech vrcholech intervalu $[-1, 1]^n$, tj. celkem 2^n lokálních minim. Dokonce se dá ukázat, že úloha kvadratického programování je NP-těžká (vzhledem k n).

Nejdříve se budeme zabývat jednodušší variantou, kdy máme omezení jen v podobě rovnic:

$$\begin{aligned} &\text{minimalizuj} && q(\mathbf{x}) \stackrel{\text{def}}{=} \tfrac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{G}\mathbf{x} + \mathbf{g}^{\mathrm{T}}\mathbf{x} \\ &\text{za předpokladu} && \mathbf{A}^{\mathrm{T}}\mathbf{x} = \mathbf{b}. \end{aligned} \tag{7.19}$$

Matice $\mathbf{A}$ zde vznikla tak, že jsme za sebe seřadili sloupcové vektory $\mathbf{a}_i$ z (7.18) (tedy $\mathbf{A} \in \mathbb{R}^{n \times m}$). Budeme předpokládat, že $\mathbf{A}$ má řád m: pokud jsou vektory $\mathbf{a}_i$ lineárně závislé, můžeme vždy ze soustavy omezení vyřadit nadbytečné rovnice.

Nejjednodušší metodou jak řešit úlohu (7.19) je přímá eliminace proměnných. Pokud si rozdělíme vektor $\mathbf{x}$ a matici $\mathbf{A}$ na části,

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix}, \qquad\qquad \mathbf{A} = \begin{pmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \end{pmatrix},$$

kde $\mathbf{x}_1 \in \mathbb{R}^m$, $\mathbf{x}_2 \in \mathbb{R}^{n-m}$ a $\mathbf{A}_1 \in \mathbb{R}^{m\times m}$ je regulární, pak rovnice z (7.19) můžeme psát jako $\mathbf{A}_1^{\mathrm{T}}\mathbf{x}_1 + \mathbf{A}_2^{\mathrm{T}}\mathbf{x}_2 = \mathbf{b}$. Odtud dostáváme

$$\mathbf{x}_1 = \mathbf{A}_1^{-\mathrm{T}}(\mathbf{b} - \mathbf{A}_2^{\mathrm{T}}\mathbf{x}_2), \tag{7.20}$$

kde $\mathbf{A}_1^{-\mathrm{T}} = \left[\mathbf{A}_1^{\mathrm{T}}\right]^{-1}$. Můžeme tedy dosadit za $\mathbf{x}_1$ lineární kombinaci vektoru $\mathbf{x}_2$. Tím získáme z $q(\mathbf{x})$ kvadratickou funkci v proměnné $\mathbf{x}_2$, $\Psi(\mathbf{x}_2)$. Řešení $\mathbf{x}_2^*$ najdeme řešením soustavy lineárních rovnic

$$\nabla\Psi(\mathbf{x}_2) = \mathbf{0} \tag{7.21}$$

a $\mathbf{x}_1^*$ pak dostaneme z (7.20).

Může se stát, že výpočet podle metody přímé eliminace je numericky nestabilní: matice $\mathbf{A}_1^{-1}$ může obsahovat příliš velká a/nebo malá čísla, která způsobí nepřesnosti při řešení soustavy (7.21). Proto se v praxi používá zobecnění eliminační metody, při kterém se v podstatě proměnné před eliminací nejdříve lineárně transformují. Tato metoda se také často označuje jako *metoda nulového prostoru.* Nechť $\mathbf{Y}$ a $\mathbf{Z}$ jsou matice typu $n \times m$ a $n \times (n-m)$ takové, že $[\mathbf{Y} : \mathbf{Z}]$ je regulární a navíc

$$\mathbf{A}^{\mathrm{T}}\mathbf{Y} = \mathbf{I}, \qquad \mathbf{A}^{\mathrm{T}}\mathbf{Z} = \mathbf{0}. \tag{7.22}$$

Pak se každé řešení rovnice $\mathbf{A}^{\mathrm{T}}\mathbf{x} = \mathbf{b}$ dá psát jako

$$\mathbf{x} = \mathbf{Y}\mathbf{b} + \boldsymbol{\delta},$$

kde $\boldsymbol{\delta}$ je nulový vektor matice $\mathbf{A}$, tj. je prvkem množiny

$$\{\boldsymbol{\delta};\ \mathbf{A}^{\mathrm{T}}\boldsymbol{\delta} = \mathbf{0}\}.$$

Tato množina bývá označována jako *nulový prostor* matice $\mathbf{A}$. Smysl matice $\mathbf{Z}$ je v tom, že její lineárně nezávislé sloupce generují nulový prostor matice $\mathbf{A}$. Množina všech řešení rovnice $\mathbf{A}^{\mathrm{T}}\mathbf{x} = \mathbf{b}$ se proto dá vyjádřit jako

$$\{\mathbf{x};\ \mathbf{x} = \mathbf{Y}\mathbf{b} + \mathbf{Z}\mathbf{y},\ \mathbf{y} \in \mathbb{R}^{n-m}\}. \tag{7.23}$$

Když dosadíme $\mathbf{x} = \mathbf{Y}\mathbf{b} + \mathbf{Z}\mathbf{y}$ v $q(\mathbf{x})$, dostaneme funkci

$$\Psi(\mathbf{y}) = \tfrac{1}{2}\mathbf{y}^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}\mathbf{G}\mathbf{Z}\mathbf{y} + (\mathbf{g} + \mathbf{G}\mathbf{Y}\mathbf{b})^{\mathrm{T}}\mathbf{Z}\mathbf{y} + \tfrac{1}{2}(\mathbf{g} + \mathbf{G}\mathbf{Y}\mathbf{b})^{\mathrm{T}}\mathbf{Y}\mathbf{b}. \tag{7.24}$$

Každé lokální minimum $\Psi(\mathbf{y})$ musí být řešením rovnice $\nabla\Psi(\mathbf{y}) = \mathbf{0}$, tj.

$$(\mathbf{Z}^{\mathrm{T}}\mathbf{G}\mathbf{Z})\mathbf{y} = -\mathbf{Z}^{\mathrm{T}}(\mathbf{g} + \mathbf{G}\mathbf{Y}\mathbf{b}). \tag{7.25}$$

Pokud je matice $\mathbf{Z}^{\mathrm{T}}\mathbf{G}\mathbf{Z} = \nabla^2\Psi(\mathbf{y})$ kladně definitní, pak podle věty 2.1 má funkce $\Psi(\mathbf{y})$ jediné minimum $\mathbf{y}^{\mathrm{T}}$ (a to je dáno rovnicí (7.25)).

Matici $\mathbf{Z}$ s vhodnými numerickými vlastnostmi můžeme získat například tak, že rozložíme matici $\mathbf{A}$ na součin

$$\mathbf{A} = \mathbf{Q}\mathbf{R},$$

kde $\mathbf{Q}$ je ortogonální matice typu $n \times n$ (podle definice je $\mathbf{Q}^{-1} = \mathbf{Q}^{\mathrm{T}}$) a $\mathbf{R}$ je horní trojúhelníková matice (tj. $r_{ij} = 0$ pro $i > j$). Jedním způsobem, jak takový rozklad vypočítat, je Householderova transformace. Součin se dá dále rozepsat jako

$$\mathbf{A} = [\mathbf{Q}_1\, \mathbf{Q}_2] \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_1\mathbf{R}_1, \tag{7.26}$$

kde $\mathbf{R}_1$ je horní trojúhelníková matice typu $m \times m$, $\mathbf{Q}_1$ je typu $n \times m$ a $\mathbf{Q}_2$ pak typu $n \times (n-m)$. Snadno se ukáže, že potom matice

$$\mathbf{Y} = \mathbf{Q}_1\mathbf{R}^{-\mathrm{T}}, \qquad \mathbf{Z} = \mathbf{Q}_2 \tag{7.27}$$

splňují požadavky metody nulového prostoru.

Úloha (7.18) v obecném tvaru se dá řešit redukcí na posloupnost problémů, ve kterých vystupují pouze rovnice. *Metoda aktivní množiny* v každém kroku vychází z přípustného bodu $\mathbf{x}^{(k)}$ a řeší problém

$$\min\{q(\mathbf{x}^{(k)} + \boldsymbol{\delta});\, \mathbf{a}_i^{\mathrm{T}}\boldsymbol{\delta} = 0,\, i \in \mathcal{A}\}, \tag{7.28}$$

kde q je účelová funkce z (7.18) a $\mathcal{A} \subseteq \mathcal{A}(\mathbf{x}^{(k)})$. Dá se ukázat, že následující jednoduchý algoritmus vede vždy k řešení, s výjimkou patologických případů. Algoritmus začíná s množinou $\mathcal{A} = \mathcal{A}(\mathbf{x}^{(1)})$.

1. Pokud $\boldsymbol{\delta} = \mathbf{0}$ je řešením (7.28), pak vypočítej Lagrangeovy činitele λ_i pro (7.28) (pokud ne, přejdi na další krok). Je-li $\lambda_i \geq 0\ \forall i$, je $\mathbf{x}^* = \mathbf{x}^{(k)}$ lokálním minimem úlohy (7.18). Jinak najdi q, pro které je $\lambda_q < 0$ a vyřaď q z $\mathcal{A}$.

2. Najdi řešení $\boldsymbol{\delta}^{(k)}$ problému (7.28).

3. Pokud $\mathbf{x}^{(k)}+\boldsymbol{\delta}^{(k)}$ je přípustný bod, polož $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}+\boldsymbol{\delta}^{(k)}$. Jinak hledej ve směru $\mathbf{x}^{(k)}+\alpha\boldsymbol{\delta}^{(k)}$, $\alpha > 0$, až dokud nenarazíš na jedno z omezení. Pak polož $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)}\boldsymbol{\delta}^{(k)}$ a do množiny $\mathcal{A}$ přidej číslo tohoto omezení (které je aktivním omezením v $\mathbf{x}^{(k+1)}$).

7.7 Metody sekvenčního kvadratického programování

Patrně nejúspěšnější metodou pro řešení obecného problému (7.1) je *metoda sekvenčního kvadratického programování* (SQP). Spočívá v převedení původní úlohy na posloupnost úloh kvadratického programování.

Označme $\mathscr{L}$ lagrangián úlohy (7.1), $\mathscr{L}(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) - \sum_i \lambda_i c_i(\mathbf{x})$. Podle věty 7.3 pro každé lokální řešení $\mathbf{x}$ platí

$$\nabla_{\mathbf{x}}\mathscr{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{0}. \tag{7.29}$$

Metoda SQP konstruuje v každé iteraci k bodu $\mathbf{x}^{(k)}$ aproximaci původního problému a z té vypočítá vektor Lagrangeových činitelů $\boldsymbol{\lambda}^{(k)}$. Pokud se metoda nachází v bodě $(\mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k)})$, snaží se metoda najít vektory $\mathbf{x}^{(k+1)}, \boldsymbol{\lambda}^{(k+1)}$, které aproximují řešení $\mathbf{x}^*$, $\boldsymbol{\lambda}^*$ původní úlohy. Pomocí Taylorovy věty můžeme gradient $\nabla\mathscr{L}(\mathbf{x}^{(k+1)}, \boldsymbol{\lambda}^{(k+1)}) = \nabla\mathscr{L}(\mathbf{x}^{(k)} + \boldsymbol{\delta}_{\mathbf{x}}, \boldsymbol{\lambda}^{(k)} + \boldsymbol{\delta}_{\boldsymbol{\lambda}})$ přibližně vyjádřit jako

$$\nabla\mathscr{L}(\mathbf{x}^{(k)} + \boldsymbol{\delta}_{\mathbf{x}}, \boldsymbol{\lambda}^{(k)} + \boldsymbol{\delta}_{\boldsymbol{\lambda}}) = \nabla_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k)}) + \left[\nabla^2\mathscr{L}(\mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k)})\right] \begin{pmatrix} \boldsymbol{\delta}_{\mathbf{x}} \\ \boldsymbol{\delta}_{\boldsymbol{\lambda}} \end{pmatrix}. \tag{7.30}$$

Protože $\nabla\mathscr{L} = \binom{\nabla_{\mathbf{x}}\mathscr{L}}{\nabla_{\boldsymbol{\lambda}}\mathscr{L}}$, je

$$\nabla^2\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)}) = \begin{bmatrix} \nabla^2_{\mathbf{x}}\mathscr{L} & \nabla_{\mathbf{x},\boldsymbol{\lambda}}\mathscr{L} \\ \nabla_{\boldsymbol{\lambda},\mathbf{x}}\mathscr{L} & \nabla^2_{\boldsymbol{\lambda}}\mathscr{L} \end{bmatrix} (\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)}) = \begin{bmatrix} \nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)}) & -\mathbf{A}^{(k)} \\ -\mathbf{A}^{(k)\mathrm{T}} & \mathbf{0} \end{bmatrix},$$

kde sloupce matice $\mathbf{A}^{(k)}$ jsou vektory $\mathbf{a}_i^{(k)} = \nabla c_i(\mathbf{x}^{(k)})$.

$$\begin{bmatrix} \nabla_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k+1)},\boldsymbol{\lambda}^{(k+1)}) \\ \nabla_{\boldsymbol{\lambda}}\mathscr{L}(\mathbf{x}^{(k+1)},\boldsymbol{\lambda}^{(k+1)}) \end{bmatrix} = \begin{bmatrix} \nabla_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)}) \\ \nabla_{\boldsymbol{\lambda}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)}) \end{bmatrix} + \begin{bmatrix} \nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)}) & -\mathbf{A}^{(k)} \\ -\mathbf{A}^{(k)\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{pmatrix} \boldsymbol{\delta}_{\mathbf{x}} \\ \boldsymbol{\delta}_{\boldsymbol{\lambda}} \end{pmatrix}. \tag{7.31}$$

Protože chceme, aby $\nabla_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k+1)},\boldsymbol{\lambda}^{(k+1)}) = \mathbf{0}$, dostáváme z (7.31) rovnici

$$\mathbf{0} = \nabla_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)}) + \nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)})\boldsymbol{\delta}_{\mathbf{x}} - \mathbf{A}^{(k)}\boldsymbol{\delta}_{\boldsymbol{\lambda}} \tag{7.32}$$

(využijeme jen první část soustavy (7.31)). Protože ale $\nabla_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)}) = \mathbf{g}^{(k)} - \mathbf{A}^{(k)}\boldsymbol{\lambda}^{(k)}$, dostáváme z (7.32)

$$\mathbf{0} = \mathbf{g}^{(k)} - \mathbf{A}^{(k)}\boldsymbol{\lambda}^{(k)} + \nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)})\boldsymbol{\delta}_{\mathbf{x}} - \mathbf{A}^{(k)}(\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}) \tag{7.33}$$

$$\mathbf{0} = \nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)})\boldsymbol{\delta}_{\mathbf{x}} + \mathbf{g}^{(k)} - \mathbf{A}^{(k)}\boldsymbol{\lambda}^{(k+1)} \tag{7.34}$$

Namísto toho, abychom přímo řešili rovnici (7.34), sestavíme optimalizační problém, který nám nejen dá řešení (7.34), ale navíc najde takové $\boldsymbol{\delta}_{\mathbf{x}}$, pro které nebude $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}_{\mathbf{x}}$ daleko od přípustné množiny. Tímto problémem je

$$\begin{array}{lll} \text{minimalizuj} & \frac{1}{2}\boldsymbol{\delta}^{\mathrm{T}}\nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)})\boldsymbol{\delta} + \mathbf{g}^{(k)\mathrm{T}}\boldsymbol{\delta} + f^{(k)} & \\ \text{za předpokladu} & \mathbf{a}_i^{\mathrm{T}}\boldsymbol{\delta} + c_i^{(k)} = 0 & i \in E \\ & \mathbf{a}_i^{\mathrm{T}}\boldsymbol{\delta} + c_i^{(k)} \geq 0 & i \in I \end{array} \tag{7.35}$$

Je-li $\mathscr{L}'$ lagrangián úlohy (7.35), pak je vidět, že

$$\nabla_{\mathbf{x}}\mathscr{L}'(\boldsymbol{\delta},\boldsymbol{\lambda}) = \nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)})\boldsymbol{\delta} + \mathbf{g}^{(k)} - \mathbf{A}^{(k)}\boldsymbol{\lambda}$$

a tedy řešení (7.35) nám dává řešení (7.34). Navíc nový bod $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}^*$ nebude příliš vzdálen od přípustné množiny původního problému (7.1), protože

$$c_i(\mathbf{x}^{(k)} + \boldsymbol{\delta}) \approx c_i(\mathbf{x}^{(k)}) + \nabla c_i(\mathbf{x}^{(k)})^{\mathrm{T}}\boldsymbol{\delta} = c_i(\mathbf{x}^{(k)}) + \mathbf{a}_i^{\mathrm{T}}\boldsymbol{\delta}. \tag{7.36}$$

Aby se matice $\nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}^{(k)},\boldsymbol{\lambda}^{(k)})$ nemusela v každé iteraci počítat od začátku, používá se obvykle její BFGS aproximace.