UNIVERZITA KOMENSKÉHO V BRATISLAVE
Fakulta matematiky, fyziky a informatiky

Numerické experimentovanie s novou parametrickou triedou kvázinewtonovských formúl.

Diplomová práca

Bc. Matúš Stanovský

Bratislava 2012

UNIVERZITA KOMENSKÉHO V BRATISLAVE

Fakulta matematiky, fyziky a informatiky

Numerické experimentovanie s novou parametrickou triedou kvázinewtonovských formúl.

Diplomová práca

Študijný odbor:	1114 Aplikovaná matematika
Študijný program:	Ekonomická a finančná matematika
Vedúci diplomovej práce:	doc. RNDr. Milan Hamala, CSc
Diplomant:	Bc. Matúš Stanovský

Bratislava 2012

65738415

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta:	Matúš Stanovský
Študijný program:	ekonomická a finančná matematika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor:	9.1.9. aplikovaná matematika
Typ záverečnej práce:	diplomová
Jazyk záverečnej práce:	slovenský

Názov: Numerickéexperimentovanie s novou parametrickou triedou kvázinewtonovských formúl.

Cieľ:
1. Experimentálne odhadnúť optimálnu hodnotu parametra v novej triede kvázi-newtonovských formúl.
2. Získanú optimálnu formulu experimentálne porovnať s BFGS formulou.

Vedúci:	doc. RNDr. Milan Hamala, CSc.
Katedra:	FMFI.KAMŠ - Katedra aplikovanej matematiky a štatistiky
Dátum zadania:	27.11.2008

Dátum schválenia: 13.02.2012

prof. RNDr. Daniel Ševčovič, CSc.
garant študijného programu

....
študent

....
vedúci práce

Prehlásenie

Čestne prehlasujem ,že som predloženú diplomovú prácu spracoval samostatne s použitím uvedenej literatúry a informačných zdrojov

Bratislava 06. 08. 2012 ....

Matúš Stanovský

Poďakovanie

Chcel by som poďakovať všetkým, ktorí mi pomáhali akýmkoľvek spôsobom pri vypracovaní tejto diplomovej práce. Najväčšia vďaka patrí môjmu diplomovému vedúcemu doc. RNDr. Milanovi Hamalovi CSc. za cenné rady, skúsenosti, množstvo času, motivácie a dobrej nálady počas celej doby, bez ktorej by som to určite nezvládol.

Abstrakt

STANOVSKÝ Matúš: *Numerické experimentovanie s novou parametrickou triedou kvázinewtonovských formúl.*
[Diplomová práca] – Univerzita Komenského v Bratislave. Fakulta matematiky, fyziky a informatiky: Katedra aplikovanej matematiky a štatistiky.
Vedúci: doc. RNDr. Milan Hamala CSc. Bratislava, 2012

Táto diplomová práca sa zaoberá problematikou minimalizácie funkcie na voľný extrém. Spomíname Newtonovu metódu a jej nedostatky a následne vznik kvázinewtonovských metód. Predstavíme si základné metódy a nakoniec nájdeme novú triedu ktorú naprogramujeme a numericky overíme jej vlastnosti.

Kľúčové slová: nelineárne programovanie, kvázinewtonovské metódy, modifikovaná PSB formula

Abstract

STANOVSKÝ Matúš: *Numerical experiments with new parametrical class of quasi-newton formula*
[Master thesis] – Comenius Univesrity in Bratislava. Faculty of Mathematics, Physics and Informatics: Department of Applied Mathematics and Statistics.
Tutor: doc. RNDr. Milan Hamala CSc. Bratislava, 2012

This master thesis take care about problem of minimization function. We descriebed Newton method and mentioned it´s disadvantages and described quasi newton methods.We remind a few famous quasi-newton methods and try to make a new parametrical class of quasi newton methods. At the end we code new formula in MATLAB and discuss about numerical quality of it.

Key words: nonlinear programming, quasi-newton method, new modified PSB formula

Obsah

Úvod

S optimalizáciou úloh sa stretneme v každom matematickom, ekonomickom a finančnom odvetví. Reálne situácie sa snažíme aproximovať pomocou rôznych matematických modelov. Nelineárne programovanie je jedným z nástrojov optimalizácie. Jeho začiatok prinieslo v polovici minulého storočia lineárne programovanie, ktoré sa snažilo opisovať rôzne javy iba pomocou lineárnych rovníc. Postupom času sa však prišlo na to, že lineárne modely sú nedostatočné pre narastajúce požiadavky praxe a tak sústredenie smerovalo na nelineárne programovanie.

Jeden z prvých postupov na riešenie úloh je Newtonova metóda, ktorá však má nedostatky(príliš veľká zložitosť, nie vždy konverguje) tak teda vznikol priestor pre iné formy. Kvazinewtonovské metódy vyriešili najväčšie nedostatky Newtonovej metódy a vzniklo niekoľko rôznych formúl.

Cieľom tejto práce je nájsť novú parametrickú triedu kvazinewtonovskej formuly a danú formulu naprogramovať a numericky zanalyzovať. Neskôr overíme jej presnosť rôznymi numerickými experimentami a na záver ju porovnáme s metódou BFGS.

Kapitola 1

Kvázinewtonovské metódy

Kvázinewtonovské metódy sú určené na riešenie takzvanej úlohy na voľný extrém.

$$Min\{f(x) \mid x \in R^n\} \qquad (U1)$$

kde účelová funkcia $f(x): \boldsymbol{R}^n \to \boldsymbol{R}$ je je vo všeobecnosti spojitá funkcia o ktorej budeme predpokladať:

- $f(x)$ je konvexná
- $f(x)$ má spojité druhé parciálne derivácie ,teda $f(x) \in C^2$

1.1. Iteračné metódy

Úlohu (U1) spravidla nevieme riešiť nejakou matematickou formulkou (predpisom) a preto sme odkázaný na iteračné metódy, ktoré postupne vylepšujú začiatočný odhad, zadaný takzvaným štartovacím bodom $\boldsymbol{x}_0 \in \boldsymbol{R}^n$, podľa schémy

$$\boldsymbol{x}_{k+1} = I(\boldsymbol{x}_k) \qquad (k = 0,1,2,\dots) \qquad (1.1)$$

Takýmto iteračným predpisom hovoríme jednoduché, alebo autonómne. Ak iteračný predpis je zložitejší

$$\boldsymbol{x}_{k+1} = I\left(\boldsymbol{x}_{k,}\boldsymbol{x}_{k-1,}, \boldsymbol{x}_{k-2,}, \dots, \boldsymbol{x}_{n,}\right) \qquad (1.2)$$

nazývame ich neautonómne, alebo tiež metódy s pamäťou.

Schémy (1.1) alebo (1.2) sa spravidla realizujú dvojfázovo. Najprv sa v bode $\boldsymbol{x}_k \in \boldsymbol{R}^n$ vytýči smer $\boldsymbol{s}_k \in \boldsymbol{R}^n$ a potom v smere $\boldsymbol{s}_k$ sa určí dĺžka kroku $\lambda_k > 0$, t.j. príslušný iteračný predpis vyzerá takto:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \lambda_k \boldsymbol{s}_k \tag{1.3}$$

1.1.1 Voľba kroku

Predpokladajme, že smer $\boldsymbol{s}_k$ je určený a λ_k zatiaľ nieje špecifikovaná. Voľba kroku λ_k je rozumná tak, aby účelová funkcia úlohy (U1) bola čo najmenšia. Teda tak ,aby

$$f[\boldsymbol{x}(\lambda)] = f(\boldsymbol{x}_k + \lambda \boldsymbol{s}_k) \tag{1.4}$$

bola minimálna. Takáto voľba λ sa nazýva *optimálna dĺžka kroku.*

1.1.2 Voľba smeru

Smer $\boldsymbol{s}_k$ zo vzťahu (1.3) môžme voliť rôznymi spôsobmi.

- V každej iteaci ho budeme voliť náhodne (stochastické algoritmy)
- Na začiatku si zvolíme systém, napríklad n-lineárne nezávislých smerov a v každej iterácií sa postupne budú voliť smery z tohto systému (napr. Relaxačná metóda)
- Smer $\boldsymbol{s}_k$ závisí od bodu $\boldsymbol{x}_k$, v ktorom sa v danej iterácií nachádzame, ako aj od minimalizovanej funkcie . (gradientné metódy)

Ďalej sa budeme zaoberať iba voľbou smeru z posledného prípadu. Konkrétne voľbou *spádového smeru*.

Majmä funkciu $f(x)$ definovanú na R^n a bod $\boldsymbol{x}_k \in R^n$, potom smer $\boldsymbol{s}_k \in R^n$ nazývame spádovým smerom funkcie $f(x)$ v bode $\boldsymbol{x}_k$, ak existuje kladné číslo $\bar{\lambda}$ tak, že pre všetky $0 < \lambda < \bar{\lambda}$ platí

$$f(\boldsymbol{x}_k + \lambda \boldsymbol{s}_k) < f(\boldsymbol{x}_k) \tag{1.5}$$

1.1.3 Cauchyho metóda najväčsieho spádu

Ak za spádový smer volíme záporný gradient $\boldsymbol{s}_k = -\nabla f(\boldsymbol{x}_k) = -\boldsymbol{g}_k$ a krok volíme optimálny, tak dostávame klasickú Cauchyho metódu najväčšieho spádu. Cauchyho metóda je konvergentná, avšak rýchlosť konvergencie nieje vo všeobecnosti uspokojivá. V prípade kvadratickej funkcie konverguje lineárne a teda vo všeobecnosti jej rýchlosť konvergencie nemôže byť lepšia ako lineárna.

1.1.4 Newtonova metóda

Uvažujme úlohu (U1) . Pre jednoduchosť gradient funkcie $\nabla f(x)$ budeme označovať $g(x)$ a jej Hessovu maticu druhých parciálnych derivácií symbolom $G(x)$ teda

$$g(x) = \nabla f(x)\,, \qquad G(x) = \nabla^2 f(x)$$

Predpokladajme, že $\boldsymbol{x}_k$ je približným riešením úlohy U(1). V okolí bodu $\boldsymbol{x}_k$ aproximujeme funkciu $f(\boldsymbol{x})$ Taylorovým polynómom druhého stupňa:

$$f(\boldsymbol{x}) \approx f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)^T(\boldsymbol{x} - \boldsymbol{x}_k) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_k)^T G(\boldsymbol{x}_k)(\boldsymbol{x} - \boldsymbol{x}_k) = Q(\boldsymbol{x}) \qquad (1.6)$$

V Newtonovej metóde sa definuje ako nové priblíženie $\boldsymbol{x}_{k+1}$ úlohy (U1) ako minimum funkcie $Q(\boldsymbol{x})$ zo vzťahu (1.6) Za predpokladu, že funkcia $f(\boldsymbol{x})$ je konvexná tak matica $G(\boldsymbol{x}_k)$ je kladne definítna. Derivovaním funkcie $Q(\boldsymbol{x})$ dostávame:

$$\nabla Q(\boldsymbol{x}) = g(\boldsymbol{x}_k) + G(\boldsymbol{x}_k)(\boldsymbol{x} - \boldsymbol{x}_k) = 0 \qquad (1.7)$$

A teda

$$(\boldsymbol{x} - \boldsymbol{x}_k) = -G(\boldsymbol{x}_k)^{-1}\, g(\boldsymbol{x}_k) \qquad (1.8)$$

respektíve

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - G(\boldsymbol{x}_k)^{-1}\, g(\boldsymbol{x}_k) \qquad (1.9)$$

Môžme si všimnúť, že spádový smer je v Newtonovej metóde definovaný ako

$$\boldsymbol{s}_k = -G(\boldsymbol{x}_k)^{-1}\, g(\boldsymbol{x}_k) \tag{1.10}$$

A teda zo vzťahov (1.3) a (1.9) vyplýva, že krok λ je konštantný a to rovný jednotke.

1.2 Čo sú to Kvázinewtonovské metódy

S použitím Newtonovej metódy vznikajú tieto problémy:

- Matica $\boldsymbol{G}_k$ nemusí byť vždy kladne definítna a teda Newtonova metóda nemusí konvergovať k minimu
- Výpočet inverznej matice $\boldsymbol{G}_k$ je časovo veľmi náročný $O(n^3)$
- Je potrebný algoritmus na výpočet $f(\boldsymbol{x}_k)$, $\boldsymbol{g}_k$ a aj $\boldsymbol{G}_k$

Všetky tieto problémy riešia Kvazinewtonovské metódy. Ako už názov napovedá, tieto metódy vychádzajú z pôvodnej Newtonovej metódy. Pri nej je funkcia f aproximovaná Taylorovým polynómom a pri hľadaní minima platí iteračný vzorec (1.8). V Kvazinewtonovských metódach sa nevyužíva inverzná Hessova matica druhých parciálnych derivácií, ale iba jej aproximácia pomocou gradientov o ktorých máme vedomosť v každom bode. Teda nieje nutné riešiť lineárny systém ako pri Newtonovej metóde.

Zoberme si kvadratickú funkciu

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\boldsymbol{G}\boldsymbol{x} + \boldsymbol{h}^T\boldsymbol{x}$$

kde $\boldsymbol{x}^*$ je bod minima a $\boldsymbol{G} > 0$. Označme $(\boldsymbol{g}_{k+1} - \boldsymbol{g}_k) = \boldsymbol{y}_k$ a $(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) = \boldsymbol{p}_k$ potom pre dva body $\boldsymbol{x}_k$ a $\boldsymbol{x}_{k+1}$ platí

$$(\boldsymbol{g}_{k+1} - \boldsymbol{g}_k) = \boldsymbol{G}\,(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) \tag{1.9}$$

$$\boldsymbol{y}_k = \boldsymbol{G}\,\boldsymbol{p}_k \quad , \quad \boldsymbol{p}_k = \boldsymbol{G}^{-1}\,\boldsymbol{y}_k \tag{1.10}$$

Vzťahy (1.10) sa nazývajú *kvázinewtonovská podmienka*. Označme aproximáciu $\boldsymbol{G}^{-1} \approx \mathbf{H}_k$ potom všeobecná schéma kvazinewtonovskej metódy vyzerá nasledovne:

1. Inicializácia: zvolíme $\boldsymbol{x}_0 \in \boldsymbol{R}^n$, $\boldsymbol{H}_0 > 0$, k = 0 a dopočítame $\boldsymbol{g}_0 = \nabla f(\boldsymbol{x}_0)$
2. Cyklus:
   a) k=0,1, ... ,n-1
   b) výpočet smeru $\boldsymbol{s}_k = -\boldsymbol{H}_k\, \boldsymbol{g}_k$
   c) výpočet optimálneho kroku $\lambda_k > 0$
   d) výpočet nového bodu $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \lambda_k * \boldsymbol{s}_k$
   e) výpočet nového gradientu $\boldsymbol{g}_{k+1} = \nabla f(\boldsymbol{x}_{k+1})$
   f) výpočet pomocných premenných $\boldsymbol{y}_k = (\boldsymbol{g}_{k+1} - \boldsymbol{g}_k)$,$\boldsymbol{p}_k = (\boldsymbol{x}_{k+1} - \boldsymbol{x}_k)$
   g) pomocou $\boldsymbol{y}_k, \boldsymbol{p}_k\ a\ \boldsymbol{H}_k$ kalkulácia $\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \Delta\boldsymbol{H}_k$
   h) k=k+1 a návrat na bod **i.**
3. Výsledok : $\boldsymbol{x}_{k-1}$

V prípade ,že kvázinewtonovská metóda nieje n-kroková tak sa bod „**a)**“ v cykle mení na TEST dosiahnutia požadovanej presnosti , teda ak bude splnená podmienka $\|\nabla f(\boldsymbol{x}_{k+1})\| < \varepsilon$ tak sa cyklus zastaví.

Podmienka na maticu $\boldsymbol{H}_0$ je, aby bola kladne definítna, teda pokiaľ úloha nemá nejakú špeciálnu štruktúru tak sa zvyčajne volí $\boldsymbol{I}$ **.** Konkrétne príklady na matice $\Delta\boldsymbol{H}_k$ uvediem v neskoršej kapitole.

1.3 História Kvazinewtonovských metód

Vznik lineárneho programovania sa datuje od roku 1947, kedy Dantzig rozpracoval simplexovú metódu riešenia úloh lineárneho programovania. Vďaka simplexovej metóde a rozvoju samočinných počítačov našlo lineárne programovanie rýchle uplatnenie v ekonomickej praxi. Tento úspech bol podmienený aj tým, že mnohé závislosti medzi ekonomickými veličinami sú blízke k lineárnym, a preto ich aproximácia lineárnymi funkciami dáva uspokojivé výsledky.

Postupne sa však formulovali nové modely, v ktorých linearizácia podstatnejšie skreslovala skutočnosť, a preto sa pozornosť začala upierať k teórii a algoritmom nelineárneho programovania. Vznik nelineárneho programovania sa datuje od roku 1950, kedy Kuhn a Tucker sformulovali a dokázali nutné podmienky pre lokálne extrémy v úlohe matematického programovania.

Kvázinewtonovské metódy začala metóda DFP (pomenovaná podľa autorov William C. Davidon, Roger Fletcher a Michael J.D. Powell) z roku 1959. Bola najprv objavená Davidonom a neskôr spopularizovaná Fletcherom a Powellom(zjedodušená v roku 1963). V roku 1965 bola objavená metóda SR1, avšak nikto nevedel kto ju objavil prvý tak sa nemôže pýšiť žiadnym prívlastkom objaviteľa. V roku 1967 prišla Broydenova trieda, z ktorej vznikla roku 1970 metóda BFGS objavená Broydenom, Fletcherom, Goldfarbom a Shannom nezávisle na sebe a k tomu každý ju objavil inou cestou. V roku 1970 prišla Greenstadt-Dennisova trieda.

.

1.4 Konkrétne Kvazinewtonovské formuly

Základná idea kvazinewtonovských metód je v zmene $\boldsymbol{H}_k$ na $\boldsymbol{H}_{k+1}$ a snaženie sa ušetriť množstvo výpočtov pri zachovaní kvázinewtonovskej podmienky (1.10) tak, aby výpočty boli relatívne „lacné“.

1.4.1 BFGS

BFGS (Broyden-Fletcher-Goldfarb-Shanno) z roku 1970 je najpopulárnejšia kvazinewtonovská metóda. Je obľúbená najmä vďaka jej perfektným konvergenčným vlastnostiam. Aktualizácia inverznej Hessovej matice $\boldsymbol{H}_{k+1}$ sa počíta ako:

$$\boldsymbol{H}_{k+1} = (\boldsymbol{I} - \boldsymbol{\alpha}_k \boldsymbol{y}_k \boldsymbol{s}_k{}^T)^T \boldsymbol{H}_k (\boldsymbol{I} - \boldsymbol{\alpha}_k \boldsymbol{y}_k \boldsymbol{p}_k{}^T) + \boldsymbol{\alpha}_k \boldsymbol{p}_k \boldsymbol{p}_k{}^T \tag{1.11}$$

kde

$$\boldsymbol{\alpha}_k = \frac{1}{\boldsymbol{y}_k^T \boldsymbol{p}_k} \tag{1.12}$$

respektíve aktualizácia Hessovej matice $\boldsymbol{B}_{k+1} = (\boldsymbol{H}_{k+1})^{-1}$

$$\boldsymbol{B}_{k+1} = \boldsymbol{B}_k + \frac{\boldsymbol{y}_k \boldsymbol{y}_k^T}{\boldsymbol{y}_k^T \boldsymbol{p}_k} - \frac{\boldsymbol{B}_k \boldsymbol{p}_k (\boldsymbol{B}_k \boldsymbol{p}_k)^T}{\boldsymbol{p}_k^T \boldsymbol{B}_k \boldsymbol{p}_k} \tag{1.13}$$

Dobre začiatočne zvolená aproximácia (inverznej) Hessovej matice sa často používa na zlepšenie výkonnosti metódy. Zvyčajne používané začiatočné voľby:

- Jednotková matica $\boldsymbol{H}_0 = \boldsymbol{I}$ alebo jej násobok $\boldsymbol{H}_0 = \beta \boldsymbol{I}$
- Vážená verzia jednotkovej matice $\boldsymbol{H}_0 = diag(s_1, s_2, \dots, s_n)$ kde s_i sú rôzne váhy
- $\boldsymbol{H}_0$ je preškálovaná ako $\boldsymbol{H}_0 = \left(\frac{\boldsymbol{y}_0^T \boldsymbol{p}_0}{\boldsymbol{y}_0^T \boldsymbol{y}_0}\right) \boldsymbol{I}$ ktoré zaručí, že matice $\boldsymbol{H}_{k+1}$ nebudú príliš veľké

1.4.2 DFP

DFP (pomenovaná podľa autorov William C. Davidon, Roger Fletcher a Michael J.D. Powell) z roku 1959 bola najprv objavená Davidonom, neskôr spopularizovaná Fletcherom a Powellom (zjedodušená v roku 1963). Je to jedna z prvých metód a považovaná za jednu z najrozumnejších pre konštrukciu inverznej Hessovej matice. Táto metóda má veľmi dobrú vlastnosť a to, že pre kvadratické účely sa súčasne generujú smery *konjungovaných gradientov,* kým sa konštruuje inverzná Hessova matica. V každom kroku je inverzná Hessova matica tvorená súčtom dvoch symetrických matíc hodnosti jedna. Prvotný názov tejto metóde dal W.C. Davidon a to „*variable metric method*“. pod ktorým sa občas vyskytuje v literatúre. Aktualizácia inverznej Hessovej matice $\boldsymbol{H}_{k+1}$ sa počíta ako:

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \frac{\boldsymbol{p}_k \boldsymbol{p}_k^T}{\boldsymbol{y}_k^T \boldsymbol{p}_k} - \frac{\boldsymbol{H}_k \boldsymbol{y}_k \boldsymbol{y}_k^T \boldsymbol{H}_k}{\boldsymbol{y}_k^T \boldsymbol{H}_k \boldsymbol{y}_k} \tag{1.14}$$

Respektíve aktualizácia Hessovej matice $\boldsymbol{B}_{k+1} = (\boldsymbol{H}_{k+1})^{-\mathbf{1}}$

$$\boldsymbol{B}_{k+1} = \left(\boldsymbol{I} - \frac{\boldsymbol{p}_k \boldsymbol{y}_k^T}{\boldsymbol{y}_k^T \boldsymbol{p}_k}\right)^T \boldsymbol{B}_k \left(\boldsymbol{I} - \frac{\boldsymbol{p}_k \boldsymbol{y}_k^T}{\boldsymbol{y}_k^T \boldsymbol{p}_k}\right) + \left(\frac{\boldsymbol{y}_k \boldsymbol{y}_k^T}{\boldsymbol{y}_k^T \boldsymbol{p}_k}\right) \tag{1.15}$$

DFP metóda je veľmi podobná BFGS metóde a keď bola objavená, tak spôsobila menšiu revolúciu na poli nelineárneho programovania.

1.4.3 SR1

SR1 (Symetric Rank 1) metóda bola predstavená v appendixe práce W.C. Davidona [1]

Aktualizácia Hessovej matice $\boldsymbol{B}_{k+1} = (\boldsymbol{H}_{k+1})^{-\mathbf{1}}$ sa počíta ako:

$$\boldsymbol{B}_{k+1} = \boldsymbol{B}_k + \left(\frac{(\boldsymbol{y}_k - \boldsymbol{B}_k \boldsymbol{p}_k)(\boldsymbol{y}_k - \boldsymbol{B}_k \boldsymbol{p}_k)^T}{(\boldsymbol{y}_k - \boldsymbol{B}_k \boldsymbol{p}_k)^T \boldsymbol{p}_k} \right) \tag{1.16}$$

Respektíve vzorec na aktualizáciu inverznej Hessovej matice $\boldsymbol{H}_{k+1}$ sa vypočíta ako:

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \left(\frac{(\boldsymbol{p}_k - \boldsymbol{H}_k \boldsymbol{y}_k)(\boldsymbol{p}_k - \boldsymbol{H}_k \boldsymbol{y}_k)^T}{(\boldsymbol{p}_k - \boldsymbol{H}_k \boldsymbol{y}_k)^T \boldsymbol{y}_k} \right) \tag{1.17}$$

Obrovská nevýhoda tejto metódy je ,že nieje zaručené, že menovateľ bude vždy kladný a teda je možné zlyhanie tejto metódy. Jej veľká výhoda je, že postupnosť aproximacií Hessových matíc je veľmi blízko reálnym Hessovym maticiam.

SR1 metóda vykazuje (n+1)-krokovosť bez nutnosti optimálneho kroku avšak posledná (n+1)-vá iterácia musí mať jednotkový krok

1.4.4 Broydenova trieda

Majmä aproximáciu Hessovej matice $\boldsymbol{B}$ a aktualizujme ju nasledovným spôsobom:

$$\boldsymbol{B}_{k+1} = \boldsymbol{B}_k + \Delta \boldsymbol{B}_k(\boldsymbol{p}_k, \boldsymbol{y}_k, \boldsymbol{B}_k) \tag{1.18}$$

Potom Broydenovu triedu kvázinewtonovských aproximácií môžme zapísať ako:

$$\Delta \boldsymbol{B}(\boldsymbol{p},\boldsymbol{y},\boldsymbol{B},\boldsymbol{\Phi}) = \frac{\boldsymbol{y}\boldsymbol{y}^T}{\boldsymbol{p}^T\boldsymbol{y}} - \frac{\boldsymbol{B}\boldsymbol{p}\boldsymbol{p}^T\boldsymbol{B}^T}{\boldsymbol{p}^T\boldsymbol{B}\boldsymbol{p}} + \boldsymbol{\Phi}(\boldsymbol{p}^T\boldsymbol{B}\boldsymbol{p})\boldsymbol{u}\boldsymbol{u}^T \tag{1.19}$$

Kde

$$\boldsymbol{u} = \frac{\boldsymbol{y}}{\boldsymbol{p}^T\boldsymbol{y}} - \frac{\boldsymbol{B}\boldsymbol{p}}{\boldsymbol{p}^T\boldsymbol{B}\boldsymbol{p}} \tag{1.20}$$

Do tejto triedy formúl patria aj vyššie spomínané formuly (DFP,BFGS,SR1) a to s nasledovnou voľbou parametra $\boldsymbol{\Phi}$:

- DFP: $\boldsymbol{\Phi} = \mathbf{1}$
- BFGS: $\boldsymbol{\Phi} = \mathbf{0}$
- SR1: $\boldsymbol{\Phi} = \frac{\boldsymbol{p}^T\boldsymbol{y}}{\boldsymbol{p}^T\boldsymbol{y} - \boldsymbol{p}^T\boldsymbol{B}\boldsymbol{p}}$

Všetky formuly z Broydenovej triedy formúl majú vlastnosť n-krokovosti

1.4.5 Greenstadt-Dennisova trieda

Ďalšou triedou kvazinewtonovských formúl je Greenstadt-Dennisova trieda. Jej predpis je nasledovný:

$$\Delta \boldsymbol{H}_{GD} = \frac{1}{\boldsymbol{d}^T\boldsymbol{y}}\left[\boldsymbol{r}\boldsymbol{d}^T + \boldsymbol{d}\boldsymbol{r}^T - \frac{\boldsymbol{y}^T\boldsymbol{r}}{\boldsymbol{d}^T\boldsymbol{y}}\boldsymbol{d}\boldsymbol{d}^T\right] \tag{1.21}$$

kde

$\boldsymbol{r} = (\boldsymbol{p} - \boldsymbol{H}\boldsymbol{y})$, $\quad \boldsymbol{d} \in \boldsymbol{R}^n$ je parameter

1.5 SR2-formula ako univerzálna trieda formúl hodnosti 2

Predpis formuly SR2 je definovaný ako:

$$\Delta \boldsymbol{H}_{SR2} = \frac{(\boldsymbol{r}+\boldsymbol{a})(\boldsymbol{r}+\boldsymbol{a})^T}{\boldsymbol{y}^T(\boldsymbol{r}+\boldsymbol{a})} - \frac{\boldsymbol{a}\boldsymbol{a}^T}{\boldsymbol{y}^T\boldsymbol{a}} \tag{1.22}$$

Kde

$$\boldsymbol{r} = (\boldsymbol{p} - \boldsymbol{H}\boldsymbol{y})\,, \qquad \boldsymbol{a} \in \boldsymbol{R}^n \text{ je parameter}$$

Táto formula je všeobecnejšia ako všeobecná Greenstadt-Dennisova formula lebo matica $\Delta\boldsymbol{H}_{GD}$ je vždy indefinítna avšak matica $\Delta\boldsymbol{H}_{SR2}$ môže byť aj kladne semidefinítna, nakoľko:

$$\boldsymbol{z}^T \Delta\boldsymbol{H}_{SR2}\boldsymbol{z} = \frac{[\boldsymbol{z}^T(\boldsymbol{r}+\boldsymbol{a})]^2}{\boldsymbol{y}^T(\boldsymbol{r}+\boldsymbol{a})} - \frac{[\boldsymbol{z}^T\boldsymbol{a}]^2}{\boldsymbol{y}^T\boldsymbol{a}} \tag{1.23}$$

voľbou parametra $\boldsymbol{a} \in \boldsymbol{R}^n$ môžme dosiahnuť:

- Ak $\boldsymbol{y}^T(\boldsymbol{r}+\boldsymbol{a}) > 0 \;\; a \;\; \boldsymbol{y}^T\boldsymbol{a} < 0$ **-** kladná semidefinítnosť
- Ak $\boldsymbol{y}^T(\boldsymbol{r}+\boldsymbol{a}) > 0 \;\; a \;\; \boldsymbol{y}^T\boldsymbol{a} < 0$ **-**záporná semidefinítnosť

Respektíve v Greenstadt-Dennisovej formulke parameter $\boldsymbol{d} \in \boldsymbol{R}^n$ **,** môžme prenásobiť ľubovoľným skalárom δ ,ale δ sa nám vykráti a teda existuje len (n-1) stupňov volnosti. V SR2 formulke je dĺžka podstatná

Majme kvadratickú funkciu $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\boldsymbol{G}\boldsymbol{x} + \boldsymbol{h}^T\boldsymbol{x}$ potom pre SR2 formulu platí nasledovná veta :

Nech postupnosť vektorov $\{\boldsymbol{a}_k\}$ má nasledovné vlastnosti:

$\boldsymbol{a}_0 \in \boldsymbol{R}^n$– jeľubovoľný vektor a

$$\boldsymbol{a}_{k+1}^T\boldsymbol{y}_i = \boldsymbol{0}\,, \forall(i = 0,1,\dots,k) \;\; a \; (i = 0,1,\dots,n-1) \tag{1.24}$$

Potom SR2 formula vykazuje nasledovné vzťahy:

a) $\boldsymbol{H}_{k+1}\boldsymbol{y}_i = \boldsymbol{p}_i$, $(i = 0,1,\dots,k)$

b) $\boldsymbol{g}_{k+1}^T\boldsymbol{p}_i = \boldsymbol{0}$, $(i = 0,1,\dots,k)$

c) $\boldsymbol{p}_i^T\boldsymbol{G}\boldsymbol{p}_{\boldsymbol{j}} = \boldsymbol{0}$, $0 \le i < j < k$

d) $\boldsymbol{r}_i^T\boldsymbol{y}_j = \boldsymbol{0}$, $\boldsymbol{r}_j^T\boldsymbol{y}_i = \boldsymbol{0}$, $0 \le i < j < k$

Z vlastnosti c) vyplýva n-krokovosť SR2

Veta bola sformulovaná a dokázaná [4]

Kapitola 2

Nová parametrická trieda kvázinewtonovskej formule

2.1 PSB metóda

Metóda vznikla v roku 1970 M. POWELL, A New Algorithm for Unconstrained Optimization, Nonlinear Programming;Proceedings, (1970) a má nasledovný predpis:

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \frac{\boldsymbol{r}\boldsymbol{y}^T + \boldsymbol{y}\boldsymbol{r}^T}{\boldsymbol{y}^T\boldsymbol{y}} - \frac{\boldsymbol{y}^T\boldsymbol{r}\boldsymbol{y}\boldsymbol{y}^T}{(\boldsymbol{y}^T\boldsymbol{y})^2} \tag{2.1}$$

Respektíve

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \frac{1}{\boldsymbol{y}^T\boldsymbol{y}}\left[\boldsymbol{y}\boldsymbol{r}^T + \boldsymbol{r}\boldsymbol{y}^T - \frac{\boldsymbol{y}^T\boldsymbol{r}}{\boldsymbol{y}^T\boldsymbol{y}}\boldsymbol{y}\boldsymbol{y}^T\right] \tag{2.2}$$

Všimnime si ,že PSB formula je špeciálnym prípadom všeobecnej Greenstadt-Dennisovej formule (1.21) ak zvolíme za parameter $\boldsymbol{d} = \boldsymbol{y}$. Numerickými experimentami sme prišli na to, že PSB formula nieje n-kroková pre účelovú funkciu v tvare

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\boldsymbol{G}\boldsymbol{x} + \boldsymbol{h}^T\boldsymbol{x}$$

n-krokovosť totiž nieje garantovaná aj vzhľadom na to, že PSB nepatrí do Broydenovej triedy.

2.2 Modifikovaná PSB metóda (mPSB)

Modifikovanú PSB metódu sme spravili ako snahu upraviť klasickú PSB metódu tak, aby vykazovala n-krokovú konvergenciu. N-krokovú konvergenciu sme nakoniec dosiahli využitím časti vety SR2 metódy konkrétne vlastnosti parametra $\boldsymbol{a}$ (1.24). Teda voľba parametra $\boldsymbol{d}$, vo všeobecnej Greenstadt-Denisovej formule (1.21) nám z numerických experimentov vychádza, ako vlastnosť (1.24) oslabená na

$$\boldsymbol{d}_{k+1}^T \boldsymbol{y}_k = \boldsymbol{0}\ , \forall (k = 0,1,\dots,n-1) \qquad (2.3.1)$$

Teda samotná voľba parametra :

$$\boldsymbol{d}_{k+1} = \boldsymbol{y}_{k+1} + \boldsymbol{\beta}_{k+1} \boldsymbol{d}_k\ , \boldsymbol{d}_0 = \boldsymbol{y}_0 \qquad (2.3.2)$$

$$\boldsymbol{\beta}_{k+1} = -\frac{\boldsymbol{y}_{k+1}^T \boldsymbol{y}_k}{\boldsymbol{d}_k^T \boldsymbol{y}_k} \qquad (2.3.3)$$

Potom n-kroková konvergencia je vykazovaná. Experimentálne sme overovali aj iné voľby parametra $\boldsymbol{d}$ ako napríklad $\boldsymbol{d}_{k+1} = \boldsymbol{p}_{k+1} + \boldsymbol{\beta}_k \boldsymbol{d}_k$, $\boldsymbol{d}_0 = \boldsymbol{p}_0$ alebo $\boldsymbol{d}_{k+1} = \boldsymbol{r}_{k+1} + \boldsymbol{\beta}_k \boldsymbol{d}_k$, $\boldsymbol{d}_0 = \boldsymbol{r}_0$Numerické experimenty ukázali najlepšie výsledky pre $\boldsymbol{d}_{k+1} = \boldsymbol{y}_{k+1} + \boldsymbol{\beta}_k \boldsymbol{d}_k$, $\boldsymbol{d}_0 = \boldsymbol{y}_0$ posledná voľba bola mierne horšia.

2.3 Interpretácia SR2 formuly ako afínnej kombinácie SR1 a GD formule

Zoberme si SR2 formulu (1.22) a upravujme

$$\Delta H_{SR2} = \frac{(r+a)(r+a)^T}{y^T(r+a)} - \frac{aa^T}{y^T a}$$

$$\Delta H_{SR2} = \frac{rr^T + ar^T + ra^T + aa^T}{(y^T r) + (y^T a)} - \frac{aa^T}{y^T a}$$

$$\Delta H_{SR2} = \frac{rr^T + ar^T + ra^T}{y^T r + y^T a} + aa^T \left[\frac{1}{y^T r + y^T a} - \frac{1}{y^T a}\right]$$

$$\Delta H_{SR2} = \frac{rr^T + ar^T + ra^T}{y^T r + y^T a} + aa^T \left[\frac{-y^T r}{y^T a(y^T r + y^T a)}\right]$$

$$\Delta H_{SR2} = \frac{rr^T}{y^T r + y^T a} + \frac{ar^T + ra^T}{y^T r + y^T a} - \frac{y^T r}{y^T a(y^T r + y^T a)} aa^T$$

$$\Delta H_{SR2} = \left(\frac{y^T r}{(y^T r + y^T a)}\right)\left(\frac{rr^T}{y^T r}\right) + \frac{1}{y^T r + y^T a}\left\{ar^T + ra^T - \frac{y^T r}{y^T a} aa^T\right\}$$

$$\Delta H_{SR2} = \left(\frac{y^T r}{(y^T r + y^T a)}\right)\left(\frac{rr^T}{y^T r}\right) + \left(\frac{y^T a}{(y^T r + y^T a)}\right)\left\{\frac{1}{y^T a}\left[ar^T + ra^T - \frac{y^T r}{y^T a} aa^T\right]\right\}$$

Zvoľme $\lambda = \left(\frac{y^T a}{(y^T r + y^T a)}\right)$ potom $(1 - \lambda) = \left(\frac{y^T r}{(y^T r + y^T a)}\right)$ a teda

$$\Delta H_{SR2} = (1 - \lambda)\left(\frac{rr^T}{y^T r}\right) + \lambda\left\{\frac{1}{y^T a}\left[ar^T + ra^T - \frac{y^T r}{y^T a} aa^T\right]\right\} \tag{2.4}$$

Čo nieje nič iné ako

$$\Delta H_{SR2} = (1 - \lambda)\Delta H_{SR1} + \lambda \Delta H_{GD} \tag{2.5}$$

Teda sme si zistili, že SR2 metóda je vlastne afínna kombinácia SR1 a GD a v prípade $0 \leq \lambda \leq 1$ hovoríme o konvexnej kombinácií.

2.4 Nová parametrická trieda kvazinewtonovskej formule ako kombinácia SR1 a mPSB

Zistenie zo state 2.3 a 2.2 nás privádza na novú parametrickú triedu kvázinewtonovskej formule, ktorú budeme ďalej uvádzať ako „Modifikovanú pseudo PSB“ so skratkou mpPSB. Z numerických experimentov sme prišli na to, že samotná SR1 konverguje n-krokovo s vynikajúcimi výsledkami. Jej prípadné zlyhanie počas experimentov nenastalo. mPSB konverguje tiež n-krokovo, preto nás bude zaujimáť ďalej ich kombinácia, teda voľba parametra v SR2 rovnakým spôsobom ako pri mPSB. . Experimentálne sme overovali aj iné voľby parametra $\boldsymbol{d}$ ako napríklad $\boldsymbol{d}_{k+1} = \boldsymbol{p}_{k+1} + \boldsymbol{\beta_k}\boldsymbol{d}_k$, $\boldsymbol{d}_0 = \boldsymbol{p}_0$, alebo $\boldsymbol{d}_{k+1} = \boldsymbol{r}_{k+1} + \boldsymbol{\beta_k}\boldsymbol{d}_k$, $\boldsymbol{d}_0 = \boldsymbol{r}_0$ Numerické experimenty ukázali najlepšie výsledky pre $\boldsymbol{d}_{k+1} = \boldsymbol{y}_{k+1} + \boldsymbol{\beta_k}\boldsymbol{d}_k$, $\boldsymbol{d}_0 = \boldsymbol{y}_0$ posledná voľba bola mierne horšia.

Teda

$$\Delta H_{mpPSB} = \frac{(r+a)(r+a)^T}{y^T(r+a)} - \frac{aa^T}{y^T a}$$

Kde

$$\boldsymbol{a}_{k+1} = \boldsymbol{y}_{k+1} + \boldsymbol{\beta_{k+1}}\boldsymbol{a}_k \quad , \qquad \boldsymbol{a}_0 = \boldsymbol{y}_0$$

$$\boldsymbol{\beta_{k+1}} = -\frac{\boldsymbol{y}_{k+1}^T\boldsymbol{y}_k}{\boldsymbol{a}_k^T\boldsymbol{y}_k}$$

Kapitola 3

Popis experimentu

V tejto kapitole sa budeme venovať aplikácií našej novej parametrickej kvazinewtonovskej formule mpPSB. Tak isto sa pozrieme aj na mPSB. Výsledky budeme porovnávať s už dobre známou BFGS metódou. Pokusy budeme robiť iba na úlohu (U1)

Úlohy budeme generovať pomocou dvoch generátorov. Prvý generátor je na kvadratické účelové funkcie a druhý je na bikvadratické účelové funkcie.

Generátory nám budú generovať matice G a D tak aby sme vopred vedeli definovať podmienenosť týchto matíc. Budeme sledovať ako číslo podmienenosti ovplyvňuje presnosť výsledku.

3.1 Ciele a prostriedky experimentu

Našim cieľom bolo experimentálne odhadnutie parametra a to sa nám ukázalo najvhodnejšia voľba $\boldsymbol{d}_{k+1}^T \boldsymbol{y}_k = \boldsymbol{0}.$ Numericky sme otestovali aj rôzne štartovacie matice a výsledky boli porovnateľné takže budeme pracovať so začiatočnou maticou $\boldsymbol{H}_0 = \boldsymbol{I}.$ Všetky numerické experimenty boli vykonávane v programovacom jazyku MATLAB konkrétne verzia 7.9.0.529 (R2009b). Výsledkami metód budeme sledovať dve hlavné hypotézy, ktoré nám pomôžu pri kvalitatívnom hodnotení našich metód a to:

- Hypotéza 1: Pri konštantnom čísle podmienenosti bude rastom rozmeru rásť chyba.

- Hypotéza 2: Pri konštantnom rozmere bude rastom čísla podmienenosti rásť chyba.

3.2 Generátory úloh

Na generovanie účelovej funkcie z úlohy (U1) sme využívalo dva generátory. Prvý generátor nám generoval kvadratické účelové funkcie. Druhý nám generoval bikvadratické funkcie.

3.2.1 Generátor kvadratických úloh

Vytvorili sme generátor kvadratických úloh s vopred zadaným optimálnym riešením $\hat{x} \in R^n$ v tvare :

$$Min\left\{ f(x) = \frac{1}{2}x^T G x + h^T x \;\middle|\; x \in R^n \right\}$$

Kde G je symetrická, kladne definítna matica rozmeru $n\ x\ n$ a $x, h \in R^n$

Vstupy generátora:

- Číslo podmienenosti matice $\kappa(G)$ (volili sme 10^2 , 10^4 , 10^6 , 10^8)
- n - rozmer úlohy (volili sme 10,50,100,200,500)
- ρ – vzdialenosť štartovacieho bodu od optimálneho riešenia $\hat{x}$ (volili sme 5,10,50,100,200,500)

Výstupy generátora:

- $\hat{x}$ - Optimálne riešenie
- G - Symetrická, kladne definítna matica rozmeru $n\ x\ n$
- h - vektor
- x_0 - štartovací bod

Postup:

- Generovanie diagonálnej matice D, ktorá má na diagonále vlastné čísla rovnomerne rozložené od 1 až po $\kappa(G)$
- Vygenerovanie pomocnej matice A, rozmeru $n\ x\ n$, ktorú rozložíme pomocou QR rozkladu a vybereme ortogonálnu maticu

- Vypočítame G maticu ako Q D Q'
- Vygenerujeme optimálne riešenie $\hat{x}$ a k nemu dopočítame štartovací bod x_0 vo vzdialenosti ρ
- Dopočítame vektor h vzhľadom na opimálne riešenie $\hat{x}$

3.2.2 Generátor bikvadratických úloh

Vytvorili sme generátor bikvadratických úloh s vopred zadaným optimálnym riešením $\hat{x} \in R^n$ v tvare :

$$Min\left\{ f(x) = \frac{1}{4}(x^T D x)^2 + \frac{1}{2}x^T G x + h^T x \;\middle|\; x \in R^n \right\}$$

Kde D je diagonálna kladne definítna matica G je symetrická, kladne definítna matica rozmeru $n\ x\ n$ a $x, h \in R^n$

Vstupy generátora:

- Číslo podmienenosti matice $\kappa(G)$ (volili sme 10^2 , 10^4 , 10^6 , 10^8)
- n - rozmer úlohy (volili sme 10,50,100,200,500)
- ρ – vzdialenosť štartovacieho bodu od optimálneho riešenia $\hat{x}$ (volili sme 5,10,50,100,200,500)

Výstupy generátora:

- $\hat{x}$ - Optimálne riešenie
- D – diagonálna, kladne definítna matica rozmeru $n\ x\ n$
- G - Symetrická, kladne definítna matica rozmeru $n\ x\ n$
- h - vektor
- x_0 - štartovací bod

Postup:

- Maticu D a maticu G získame rovnakým postupom ako pri predchádzajúcom generátore
- Generujeme optimálne riešenie $\hat{x}$ a k nemu dopočítame štartovací bod x_0 vo vzdialenosti ρ
- Dopočítame vektor h vzhľadom na opimálne riešenie $\hat{x}$

3.3 Štruktúra programu v MATLABE

Počas experimentov sme využívali niekoľko programov, ktoré nám poskytovali numerické výsledky metód, ktoré sme skúmali. Numerický experiment pre každú úlohu (kvadratickú,bikvadratickú) sme pre každú voľbu čísla podmienenosti, rozmeru a vzdialenosti štartovacieho bodu vykonali 50 krát. Výsledok ktorý je uvádzaný sme nakoniec spriemerovali. V zdrojovom kóde nazývam funkcie ktoré riešia kvadratickú účelovú funkcie ako BFGS , mPSB4 a mpPSB. Funkcie ktoré riešia bikvadratickú formu účelovej funkcie nazývame bBFGS, bmPSB4 a bmpPSB.

3.3.1 Kvadratické úlohy

Kvadratické úlohy sme riešili pomocou troch metód: BFGS, mPSB, mpPSB. Každá metóda mala rovnaké vstupy ktoré boli z generátora zo state 3.2.1 a to

- Optimálne riešenie $\hat{x}$ a začiatočný bod x_0
- G - Symetrická, kladne definítna matica, rozmeru $n\,x\,n$ a k nej prislúchajúci vektor h , vzhľadom na optimálne riešenie $\hat{x}$

Jej výstupom bola vždy odchýlka poslednej iterácie od optimálneho riešenia. Telo cyklu bolo zostavené podľa schémy, ktorá je uvedená v stati 1.2, kde krok λ bol vždy optimálnym krokom počítaným ako

$$\lambda = -\frac{\boldsymbol{g}_k^T \boldsymbol{s}_k}{\boldsymbol{s}_k^T \boldsymbol{G} \boldsymbol{s}_k}$$

Gradient funkcie bol počítaný ako $\boldsymbol{g}_k = \nabla f(\boldsymbol{x}_k) = \boldsymbol{G}\boldsymbol{x}_k + \boldsymbol{h}$

Príslušné aproximácie inverzných Hessových matíc boli počítané predpismi ktoré sme vyššie uviedli.

V prípade mPSB a mpPSB bola voľba parametra $\boldsymbol{d}_{k+1} = \boldsymbol{y}_{k+1} + \boldsymbol{\beta}_k \boldsymbol{d}_k$, $\boldsymbol{d}_0 = \boldsymbol{y}_0$ a $\boldsymbol{a}_{k+1} = \boldsymbol{y}_{k+1} + \boldsymbol{\beta}_k \boldsymbol{a}_k$, $\boldsymbol{a}_0 = \boldsymbol{y}_0$

3.3.2 Bikvadratické úlohy

Bikvadratické úlohy sme riešili pomocou troch metód: bBFGS, bmPSB, bmpPSB. Každá metóda mala rovnaké vstupy ktoré boli z generátora zo state 3.2.2 a to

- Optimálne riešenie $\hat{x}$ a začiatočný bod x_0
- D – diagonálna, kladne definítna matica, rozmeru $n\ x\ n$, G - Symetrická, kladne definítna matica rozmeru $n\ x\ n$ a k nim prislúchajúci vektor h vzhľadom na opimálne riešenie $\hat{x}$
- ZR – ako konštanta na voľbu počtu krokov zlatého rezu pri výpočte oprimálneho kroku

Jej výstupom bola vždy odchýlka poslednej iterácie od optimálneho riešenia. Telo cyklu bolo zostavené podľa schémy, ktorá je uvedená v stati 1.2 , kde gradient funkcie bol počítaný ako $\boldsymbol{g}_k = \nabla f(\boldsymbol{x}_k) = ((\boldsymbol{x}_k^T \boldsymbol{D} \boldsymbol{x}_k)\boldsymbol{D} + \boldsymbol{G})\boldsymbol{x}_k + \boldsymbol{h}$. Príslušné aproximácie inverzných Hessových matíc boli počítané predpismi, ktoré sme vyššie uviedli.

Optimálny krok λ v prípade bikvadratických funkcií nevieme riešiť presnou formulkou, tak na jeho výpočet sme využívali metódu zlatého rezu(ZR2), ktorý fungoval klasickým predpisom. Počet iterácií zlatého rezu sme volili väčšie ako zvyčajne, nakoľko smer $\boldsymbol{s}_k$ nám počas experimentov vychádzal často príliš veľký a potrebovali sme čo najväčšiu presnosť veľkosti kroku.

V prípade bmPSB a bmpPSB bola voľba parametra $\boldsymbol{d}_{k+1} = \boldsymbol{y}_{k+1} + \boldsymbol{\beta}_k \boldsymbol{d}_k$, $\boldsymbol{d}_0 = \boldsymbol{y}_0$ a $\boldsymbol{a}_{k+1} = \boldsymbol{y}_{k+1} + \boldsymbol{\beta}_k \boldsymbol{a}_k$, $\boldsymbol{a}_0 = \boldsymbol{y}_0$

Kapitola 4

Výsledky experimentu

V tejto kapitole si ukážeme numerické výsledky sledovaných metód pre niektoré vybrané hodnoty parametrov : Číslo podmienenosti matíc $\kappa(G)$, respektíve $\kappa(D)$, vzdialenosť štartovacieho bodu $\boldsymbol{x}_0$ od optimálneho riešenia $\hat{x}$ a rozmeru n . Výsledky nových kvazinewtonovských metód mPSB a mpPSB v porovnaní s metódou BFGS sú uvedené v tabuľkách. Kompletné numerické výsledky sú zahrnuté v prílohách tejto práce

4.1 Kvadratické úlohy

4.1.1 Hypotéza 1

- Hypotéza 1 :Pri konštantnom čísle podmienenosti bude rastom rozmeru rásť chyba

Tabulka1.1 Experiment pre $\kappa(G) = 10^2$

II Xopt-X0 II=50		IIXaprox-XoptII	
Rozmer	mpPSB	pPSB	BFGS
10	7,02E+00	3,38E-13	6,84E-04
50	3,02E-01	1,07E-11	3,42E-12
100	2,67E-02	1,34E-10	1,08E-12
200	2,06E-04	1,31E-10	1,83E-12
300	2,38E-06	3,99E-12	2,58E-12

Tabuľka 1.2 Experiment pre $\kappa(G) = 10^4$

II Xopt-X0 II=100		IIXaprox-XoptII	
Rozmer	mpPSB	pPSB	BFGS
10	2,54E+01	4,63E-11	2,28E-07
50	1,16E+01	2,05E-03	1,83E-08
100	6,98E+00	5,59E-02	2,24E-10
200	4,16E+00	2,56E-01	3,85E-10
300	3,87E+00	4,86E-01	5,24E-10

Tabuľka 1.3 Experiment pre $\kappa(G) = 10^6$

II Xopt-X0 II=50		IIXaprox-XoptII	
Rozmer	mpPSB	pPSB	BFGS
10	1,05E+01	3,96E-09	1,19E-08
50	6,01E+00	8,42E-03	2,05E-08
100	4,21E+00	5,42E-02	2,53E-08
200	2,52E+00	1,42E-01	2,30E-08
300	2,23E+00	2,47E-01	2,77E-08

Tabuľka 1.4 Experiment pre $\kappa(G) = 10^8$

II Xopt-X0 II=5		IIXaprox-XoptII	
Rozmer	mpPSB	pPSB	BFGS
10	1,18E+00	3,65E-07	7,47E-07
50	5,30E-01	1,25E-01	1,29E-06
100	4,19E-01	2,59E-01	1,56E-05
200	3,11E-01	2,86E-01	1,20E-05
300	2,30E-01	2,16E-01	6,63E-06

4.1.2 Hypotéza 2

- Hypotéza 2: Pri konštantnom rozmere bude rastom čísla podmienenosti rásť chyba

Tabuľka 1.5 Experiment pre rozmer $n = 10$

II Xopt-X0 II=500	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	6,31E+01	4,21E-13	6,35E-03
10^4	1,00E+02	4,08E-11	1,12E-06
10^6	1,32E+02	3,44E-09	1,04E-07
10^8	1,44E+02	2,99E-07	1,11E-05

Tabuľka 1.6 Experiment pre rozmer $n = 50$

II Xopt-X0 II=50	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	3,02E-01	1,07E-11	3,42E-12
10^4	5,31E+00	2,91E-04	4,94E-09
10^6	6,01E+00	8,42E-03	2,05E-08
10^8	5,37E+00	1,18E-01	2,83E-06

Tabuľka 1.7 Experiment pre rozmer $n = 100$

II Xopt-X0 II=50	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	2,67E-02	1,34E-10	1,08E-12
10^4	2,72E+00	1,96E-02	2,61E-10
10^6	4,21E+00	5,42E-02	2,53E-08
10^8	4,07E+00	7,46E-01	8,95E-07

Tabuľka 1.8 Experiment pre rozmer $n = 300$

II Xopt-X0 II=5	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	2,54E-07	3,62E-12	2,61E-12
10^4	1,89E-01	3,18E-02	3,15E-10
10^6	2,32E-01	3,61E-02	2,79E-08
10^8	2,30E-01	2,16E-01	6,63E-06

4.2 Bikvadratické úlohy

4.2.1 Hypotéza 1

- Hypotéza 1 :Pri konštantnom čísle podmienenosti bude rastom rozmeru rásť chyba

Tabuľka 2.1 Experiment pre $\kappa(G) = 10^2$

II Xopt-X0 II=100	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	1,53E+01	1,18E+01	3,71E+00
50	6,45E-01	3,02E-01	1,91E-06
100	4,67E-02	1,71E-03	3,48E-06
200	4,85E-02	4,70E-06	5,85E-06
300	5,87E-04	5,64E-06	8,34E-06

Tabuľka 2.2 Experiment pre $\kappa(G) = 10^4$

II Xopt-X0 II=100	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	3,24E+01	3,08E+01	4,65E+01
50	1,15E+01	9,57E+00	1,30E+00
100	8,78E+00	7,49E+00	2,34E-02
200	7,42E+00	5,16E+00	1,48E-05
300	5,77E+00	4,01E+00	1,04E-05

Tabuľka 2.3 Experiment pre $\kappa(G) = 10^6$

II Xopt-X0 II=500	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	7,35E+01	1,19E+02	8,64E+01
50	5,80E+01	5,28E+01	5,43E+01
100	3,16E+01	2,42E+01	3,64E+01
200	9,47E+00	1,79E+01	2,06E-01
300	6,68E+00	6,38E+00	2,67E-04

Tabuľka 2.4 Experiment pre $\kappa(G) = 10^8$

‖Xopt-X0‖=50	‖Xaprox-Xopt‖		
rozmer	bmpPSB	bpPSB	bBFGS
10	1,49E+01	1,29E+01	1,74E+01
50	6,49E+00	5,98E+00	1,82E+01
100	4,42E+00	3,78E+00	4,97E+00
200	3,00E+00	2,62E+00	2,74E+00
300	1,96E+00	1,69E+00	8,67E+00

4.2.2 Hypotéza 2

- Hypotéza 2: Pri konštantnom rozmere bude rastom čísla podmienenosti rásť chyba

Tabuľka 2.5 Experiment pre rozmer $n = 10$

‖Xopt-X0‖=100	‖Xaprox-Xopt‖		
kappa	mpPSB	pPSB	BFGS
10^2	1,53E+01	1,18E+01	3,71E+00
10^4	3,24E+01	3,08E+01	4,65E+01
10^6	1,78E+01	1,43E+01	1,42E+01
10^8	2,26E+01	2,09E+01	2,73E+01

Tabuľka 2.6 Experiment pre rozmer $n = 50$

‖Xopt-X0‖=10	‖Xaprox-Xopt‖		
kappa	mpPSB	pPSB	BFGS
10^2	1,02E-01	4,46E-02	1,75E-06
10^4	1,29E+00	1,17E+00	6,18E-03
10^6	1,45E+00	1,41E+00	2,57E+00
10^8	1,40E+00	1,31E+00	2,94E+00

Tabuľka 2.7 Experiment pre rozmer $n = 100$

‖Xopt-X0‖=50	‖Xaprox-Xopt‖		
kappa	mpPSB	pPSB	BFGS
10^2	7,88E-02	6,73E-04	3,33E-06
10^4	4,51E+00	3,46E+00	2,74E-03
10^6	5,62E+00	4,99E+00	3,15E-01
10^8	4,42E+00	3,78E+00	4,97E+00

Tabuľka 2.8 Experiment pre rozmer $n = 300$

II Xopt-X0 II=5	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	2,11E-05	6,11E-06	9,11E-06
10^4	2,42E-01	1,78E-01	1,03E-05
10^6	2,91E-01	2,62E-01	1,40E-05
10^8	2,76E-01	2,30E-01	2,00E-01

Záver

V tejto práci sme sa venovali kvázinewtonovským metódam. Najprv sme si spomenuli na Newtonovui metódu a jej hlavné nevýhody, ktoré boli tiež dôvodom vzniku inej formy optimalizácie, ktorú sme našli v aproximovaní Hessovej matice a teda Kvazinewtonovským metódam. Sformulovali sme si všeobecný algoritmus kvazinewtonovských metód, podľa ktorého sme neskôr aj programovali.

Spomenuli sme si konkrétne kvázinewtonovské formule a ich formy výpočtu aproximovaného Hessianu. Spomenuli sme Broydenovu triedu a ako z nej sa dajú odvodiť niektoré formule. Ďalej sme si spomenuli Greenstadt-Dennisovu triedu a ako sa správnou voľbou parametra odvodí PSB formula. Zistili sme, že PSB metóda nekoverguje za n-krokov, pre kvadratické účelové funkcie, tak sme ju trošku poopravili a získali pomocou jej modifikácie n–krokovosť, ktorú sme si tiež neskôr overili ako aj jej konvergenciu a n-krokovosť

Nakoniec sme si odvodili aj našu druhú novú metódu, ako kombináciu našej novej modifikovanej PSB (mPSB)(ako správnu voľbu parametra v GD) a SR1 metódy. Tieto metódy sme v poslednej časti porovnávali s BFGS metódou.

Výsledky nám vyvrátili hypotézu o zväčšovaní chyby pri konštantnom čísle podmienenosti a zvačšovaním rozmeru. Výsledky boli opačné. Hypotézu o zväčšovaní chyby pri konštantnom rozmere a zväčšovaní čísla podmienenosti sa nám podarilo potvrdiť. Numerické výsledky boli jasné.

Celkovo môžme povedať, že naša mpPSB dopadla najhoršie. Jej konvergencia výrazne zaostáva za BFGS ako aj mPSB. BFGS je oproti mPSB lepšie konvergentná vo väčších rozmeroch ako aj pri väčších číslach podmienenosti. mPSB je oproti BFGS lepšie konvergentná v menších číslach podmienenosti a menších rozmeroch. Otázkou do budúcnosti ostáva prečo nám stačili takéto(slabšie 2.3.1, 2.3.2) podmienky na parameter?

Literatúra

[1] W. C. DAVIDON, Variable Metric Method For Minimization, Technical Paper ANL-5990, Argonne National Laboratory/University of Chicago, Lemont, Illinois, (1959).

[2] E. Spedicato, A variable-metric method for function minimization derived from invariancy to nonlinear scaling, Journal of optimization theory and applications: Vol. 20, No. 3 (1976), p. 315-329.

[3] M. Hamala Nelineárne programovanie, ALFA, Bratislava.. (1972)

[4] M. Hamala, Nelineárne programovanie, konspekty prednášok, Univerzita Komenského, Bratislava, (2000).

[5] John Greenstadt ,A Quasi-Newton Method with No Derivatives , MATHEMATICS OF COMPUTATION, VOLUME 26, NUMBER 117, JANUARY, (1972)

[6] A.R.Conn, N.I.M.Gould, Ph.L.Toint, Mathematical Programming 50 (1991)

[7] C. G. Broyden, "Quasi-Newton methods and their application to function minimisation,"Math. Comp., v. 21, (1967), pp. 368-381.

[8] David G. Luenberger ,Yinyu Ye Linear and Nonlinear Programming,third edition (2008)

[9] Conn-Gould-Toint, Convergence of QN Matrices generated by the SR1 Update (1991)

[10] Sun,The Convergence of QN Matrices Generated by the Self-Scaling SR1 Update (1998)

[11] Papakonstantinou Historical Development of the BFGS Secant Method (2009)

[12] Malmedy-TointApproximating Hessians in Multilevel Unconstrained Optimization(2008)

Príloha (najprv kompletné numerické experimenty)

Hypo1 zafixovanie kappa - rastom rozmeru rastie chyba **Kappa=10^2**

II Xopt-X0 II=5 — IIXaprox-XoptII

rozmer	mpPSB	pPSB	BFGS
10	6,12E-01	3,37E-13	5,90E-05
50	2,59E-02	1,88E-11	1,05E-12
100	2,29E-03	8,94E-11	9,08E-13
200	2,16E-05	3,80E-11	1,84E-12
300	2,54E-07	3,62E-12	2,61E-12

II Xopt-X0 II=10 — IIXaprox-XoptII

rozmer	mpPSB	pPSB	BFGS
10	1,20E+00	3,18E-13	1,28E-04
50	6,23E-02	1,71E-11	1,21E-12
100	5,38E-03	8,62E-11	9,80E-13
200	4,45E-05	5,96E-11	1,87E-12
300	5,18E-08	3,70E-12	3,01E-12

II Xopt-X0 II=50 — IIXaprox-XoptII

rozmer	mpPSB	pPSB	BFGS
10	7,02E+00	3,38E-13	6,84E-04
50	3,02E-01	1,07E-11	3,42E-12
100	2,67E-02	1,34E-10	1,08E-12
200	2,06E-04	1,31E-10	1,83E-12
300	2,38E-06	3,99E-12	2,58E-12

II Xopt-X0 II=100 — IIXaprox-XoptII

rozmer	mpPSB	pPSB	BFGS
10	1,37E+01	3,33E-13	1,81E-03
50	5,81E-01	1,37E-11	6,30E-12
100	5,11E-02	1,23E-10	1,14E-12
200	4,89E-04	1,71E-10	1,85E-12
300	6,37E-06	4,48E-12	2,73E-12

II Xopt-X0 II=200 — IIXaprox-XoptII

rozmer	mpPSB	pPSB	BFGS
10	1,45E+01	3,48E-13	4,15E-03
50	1,26E+00	2,27E-11	1,47E-11
100	1,02E-01	8,84E-11	1,02E-12
200	7,97E-04	2,51E-10	1,86E-12
300	8,36E-06	4,89E-12	2,70E-12

II Xopt-X0 II=500 — IIXaprox-XoptII

rozmer	mpPSB	pPSB	BFGS
10	6,31E+01	4,21E-13	6,35E-03
50	3,60E+00	2,97E-11	2,90E-11
100	2,63E-01	9,28E-11	9,95E-13
200	2,51E-03	3,57E-10	1,93E-12
300	2,66E-05	6,19E-12	2,57E-12

Hypo1 zafixovanie kappa - rastom rozmeru rastie chyba **Kappa=10^4**

II Xopt-X0 II=5

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	1,28E+00	3,60E-11	1,08E-08
50	4,88E-01	3,01E-04	4,46E-10
100	3,26E-01	6,99E-03	3,29E-10
200	2,42E-01	1,96E-02	2,56E-10
300	1,89E-01	3,18E-02	3,15E-10

II Xopt-X0 II=10

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	2,59E+00	2,81E-11	1,62E-08
50	1,06E+00	4,14E-04	1,17E-09
100	7,00E-01	9,05E-03	3,20E-10
200	4,93E-01	3,75E-02	3,91E-10
300	4,13E-01	6,37E-02	3,40E-10

II Xopt-X0 II=50

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	1,09E+01	3,30E-11	2,11E-07
50	5,31E+00	2,91E-04	4,94E-09
100	2,72E+00	1,96E-02	2,61E-10
200	2,67E+00	1,72E-01	5,45E-10
300	1,80E+00	2,31E-01	4,11E-10

II Xopt-X0 II=100

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	2,54E+01	4,63E-11	2,28E-07
50	1,16E+01	2,05E-03	1,83E-08
100	6,98E+00	5,59E-02	2,24E-10
200	4,16E+00	2,56E-01	3,85E-10
300	3,87E+00	4,86E-01	5,24E-10

II Xopt-X0 II=200

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	5,19E+01	4,41E-11	7,75E-07
50	1,81E+01	5,51E-04	1,75E-08
100	1,31E+01	6,14E-02	1,95E-10
200	9,29E+00	4,24E-01	2,97E-10
300	6,42E+00	7,81E-01	3,00E-10

II Xopt-X0 II=500

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	1,00E+02	4,08E-11	1,12E-06
50	5,86E+01	1,36E-04	6,11E-08
100	3,82E+01	2,93E-01	1,52E-10
200	2,52E+01	1,23E+00	1,83E-10
300	1,79E+01	1,83E+00	2,01E-10

Hypo1 zafixovanie kappa - rastom rozmeru rastie chyba **Kappa=10^6**

II Xopt-X0 II=5

rozmer	mpPSB	IIXaprox-XoptII pPSB	BFGS
10	1,28E+00	3,73E-09	9,11E-09
50	6,58E-01	2,31E-03	1,51E-08
100	3,99E-01	8,39E-03	2,51E-08
200	3,17E-01	2,81E-02	2,47E-08
300	2,32E-01	3,61E-02	2,79E-08

II Xopt-X0 II=10

rozmer	mpPSB	IIXaprox-XoptII pPSB	BFGS
10	2,70E+00	2,93E-09	8,64E-09
50	1,03E+00	2,90E-03	1,20E-08
100	8,36E-01	1,43E-02	3,60E-08
200	4,80E-01	4,31E-02	3,19E-08
300	4,74E-01	7,41E-02	2,95E-08

II Xopt-X0 II=50

rozmer	mpPSB	IIXaprox-XoptII pPSB	BFGS
10	1,05E+01	3,96E-09	1,19E-08
50	6,01E+00	8,42E-03	2,05E-08
100	4,21E+00	5,42E-02	2,53E-08
200	2,52E+00	1,42E-01	2,30E-08
300	2,23E+00	2,47E-01	2,77E-08

II Xopt-X0 II=100

rozmer	mpPSB	IIXaprox-XoptII pPSB	BFGS
10	2,46E+01	3,14E-09	2,44E-08
50	9,94E+00	1,23E-02	3,01E-08
100	8,48E+00	8,06E-02	1,30E-08
200	6,16E+00	2,74E-01	2,33E-08
300	4,75E+00	4,44E-01	2,76E-08

II Xopt-X0 II=200

rozmer	mpPSB	IIXaprox-XoptII pPSB	BFGS
10	6,11E+01	3,10E-09	4,24E-08
50	2,09E+01	2,28E-02	6,03E-08
100	2,08E+01	2,12E-01	1,11E-08
200	1,13E+01	5,16E-01	1,85E-08
300	8,93E+00	7,67E-01	2,22E-08

II Xopt-X0 II=500

rozmer	mpPSB	IIXaprox-XoptII pPSB	BFGS
10	1,32E+02	3,44E-09	1,04E-07
50	6,00E+01	4,40E-02	1,46E-07
100	4,73E+01	4,72E-01	9,04E-09
200	2,13E+01	7,73E-01	1,79E-08
300	2,07E+01	1,48E+00	1,71E-08

Hypo1 zafixovanie kappa - rastom rozmeru rastie chyba **Kappa=10^8**

II Xopt-X0 II=5

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	1,18E+00	3,65E-07	7,47E-07
50	5,30E-01	1,25E-01	1,29E-06
100	4,19E-01	2,59E-01	1,56E-05
200	3,11E-01	2,86E-01	1,20E-05
300	2,30E-01	2,16E-01	6,63E-06

II Xopt-X0 II=10

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	2,38E+00	3,55E-07	8,20E-07
50	9,90E-01	1,24E-01	1,41E-06
100	8,77E-01	3,99E-01	3,19E-06
200	5,28E-01	4,13E-01	2,20E-05
300	6,11E-01	5,39E-01	1,62E-05

II Xopt-X0 II=50

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	1,30E+01	3,26E-07	1,29E-06
50	5,37E+00	1,18E-01	2,83E-06
100	4,07E+00	7,46E-01	8,95E-07
200	2,66E+00	1,22E+00	2,11E-05
300	2,40E+00	1,60E+00	1,38E-05

II Xopt-X0 II=100

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	2,17E+01	3,19E-07	2,48E-06
50	1,19E+01	7,49E-02	4,95E-06
100	8,23E+00	7,86E-01	7,79E-07
200	6,62E+00	2,12E+00	2,27E-05
300	5,04E+00	2,70E+00	1,67E-05

II Xopt-X0 II=200

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	5,68E+01	3,17E-07	4,90E-06
50	2,41E+01	8,82E-02	1,06E-05
100	1,76E+01	9,16E-01	9,83E-07
200	1,08E+01	2,71E+00	2,36E-05
300	9,53E+00	3,64E+00	9,99E-06

II Xopt-X0 II=500

		IIXaprox-XoptII	
rozmer	mpPSB	pPSB	BFGS
10	1,44E+02	2,99E-07	1,11E-05
50	5,83E+01	1,14E-01	2,36E-05
100	3,39E+01	1,31E+00	7,99E-07
200	2,85E+01	4,17E+00	2,15E-05
300	2,40E+01	5,56E+00	1,10E-05

Hypo2 zafixovanie rozmeru- rastom kappa rastie chyba **Rozmer=10**

II Xopt-X0 II=5		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	6,12E-01	3,37E-13	5,90E-05
10^4	1,28E+00	3,60E-11	1,08E-08
10^6	1,28E+00	3,73E-09	9,11E-09
10^8	1,18E+00	3,65E-07	7,47E-07

II Xopt-X0 II=10		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	1,20E+00	3,18E-13	1,28E-04
10^4	2,59E+00	2,81E-11	1,62E-08
10^6	2,70E+00	2,93E-09	8,64E-09
10^8	2,38E+00	3,55E-07	8,20E-07

II Xopt-X0 II=50		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	7,02E+00	3,38E-13	6,84E-04
10^4	1,09E+01	3,30E-11	2,11E-07
10^6	1,05E+01	3,96E-09	1,19E-08
10^8	1,30E+01	3,26E-07	1,29E-06

II Xopt-X0 II=100		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	1,37E+01	3,33E-13	1,81E-03
10^4	2,54E+01	4,63E-11	2,28E-07
10^6	2,46E+01	3,14E-09	2,44E-08
10^8	2,17E+01	3,19E-07	2,48E-06

II Xopt-X0 II=200		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	1,45E+01	3,48E-13	4,15E-03
10^4	5,19E+01	4,41E-11	7,75E-07
10^6	6,11E+01	3,10E-09	4,24E-08
10^8	5,68E+01	3,17E-07	4,90E-06

II Xopt-X0 II=500		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	6,31E+01	4,21E-13	6,35E-03
10^4	1,00E+02	4,08E-11	1,12E-06
10^6	1,32E+02	3,44E-09	1,04E-07
10^8	1,44E+02	2,99E-07	1,11E-05

Hypo2 zafixovanie rozmeru- rastom kappa rastie chyba **Rozmer=50**

II Xopt-X0 II=5 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	2,59E-02	1,88E-11	1,05E-12
10^4	4,88E-01	3,01E-04	4,46E-10
10^6	6,58E-01	2,31E-03	1,51E-08
10^8	5,30E-01	1,25E-01	1,29E-06

II Xopt-X0 II=10 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	6,23E-02	1,71E-11	1,21E-12
10^4	1,06E+00	4,14E-04	1,17E-09
10^6	1,03E+00	2,90E-03	1,20E-08
10^8	9,90E-01	1,24E-01	1,41E-06

II Xopt-X0 II=50 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	3,02E-01	1,07E-11	3,42E-12
10^4	5,31E+00	2,91E-04	4,94E-09
10^6	6,01E+00	8,42E-03	2,05E-08
10^8	5,37E+00	1,18E-01	2,83E-06

II Xopt-X0 II=100 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	5,81E-01	1,37E-11	6,30E-12
10^4	1,16E+01	2,05E-03	1,83E-08
10^6	9,94E+00	1,23E-02	3,01E-08
10^8	1,19E+01	7,49E-02	4,95E-06

II Xopt-X0 II=200 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	1,26E+00	2,27E-11	1,47E-11
10^4	1,81E+01	5,51E-04	1,75E-08
10^6	2,09E+01	2,28E-02	6,03E-08
10^8	2,41E+01	8,82E-02	1,06E-05

II Xopt-X0 II=500 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	3,60E+00	2,97E-11	2,90E-11
10^4	5,86E+01	1,36E-04	6,11E-08
10^6	6,00E+01	4,40E-02	1,46E-07
10^8	5,83E+01	1,14E-01	2,36E-05

Hypo2 zafixovanie rozmeru- rastom kappa rastie chyba **Rozmer=100**

II Xopt-X0 II=5 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	2,29E-03	8,94E-11	9,08E-13
10^4	3,26E-01	6,99E-03	3,29E-10
10^6	3,99E-01	8,39E-03	2,51E-08
10^8	4,19E-01	2,59E-01	1,56E-05

II Xopt-X0 II=10 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	5,38E-03	8,62E-11	9,80E-13
10^4	7,00E-01	9,05E-03	3,20E-10
10^6	8,36E-01	1,43E-02	3,60E-08
10^8	8,77E-01	3,99E-01	3,19E-06

II Xopt-X0 II=50 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	2,67E-02	1,34E-10	1,08E-12
10^4	2,72E+00	1,96E-02	2,61E-10
10^6	4,21E+00	5,42E-02	2,53E-08
10^8	4,07E+00	7,46E-01	8,95E-07

II Xopt-X0 II=100 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	5,11E-02	1,23E-10	1,14E-12
10^4	6,98E+00	5,59E-02	2,24E-10
10^6	8,48E+00	8,06E-02	1,30E-08
10^8	8,23E+00	7,86E-01	7,79E-07

II Xopt-X0 II=200 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	1,02E-01	8,84E-11	1,02E-12
10^4	1,31E+01	6,14E-02	1,95E-10
10^6	2,08E+01	2,12E-01	1,11E-08
10^8	1,76E+01	9,16E-01	9,83E-07

II Xopt-X0 II=500 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	2,63E-01	9,28E-11	9,95E-13
10^4	3,82E+01	2,93E-01	1,52E-10
10^6	4,73E+01	4,72E-01	9,04E-09
10^8	3,39E+01	1,31E+00	7,99E-07

Hypo2 zafixovanie rozmeru- rastom kappa rastie chyba **Rozmer=300**

II Xopt-X0 II=5 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	2,54E-07	3,62E-12	2,61E-12
10^4	1,89E-01	3,18E-02	3,15E-10
10^6	2,32E-01	3,61E-02	2,79E-08
10^8	2,30E-01	2,16E-01	6,63E-06

II Xopt-X0 II=10 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	5,18E-08	3,70E-12	3,01E-12
10^4	4,13E-01	6,37E-02	3,40E-10
10^6	4,74E-01	7,41E-02	2,95E-08
10^8	6,11E-01	5,39E-01	1,62E-05

II Xopt-X0 II=50 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	2,38E-06	3,99E-12	2,58E-12
10^4	1,80E+00	2,31E-01	4,11E-10
10^6	2,23E+00	2,47E-01	2,77E-08
10^8	2,40E+00	1,60E+00	1,38E-05

II Xopt-X0 II=100 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	6,37E-06	4,48E-12	2,73E-12
10^4	3,87E+00	4,86E-01	5,24E-10
10^6	4,75E+00	4,44E-01	2,76E-08
10^8	5,04E+00	2,70E+00	1,67E-05

II Xopt-X0 II=200 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	8,36E-06	4,89E-12	2,70E-12
10^4	6,42E+00	7,81E-01	3,00E-10
10^6	8,93E+00	7,67E-01	2,22E-08
10^8	9,53E+00	3,64E+00	9,99E-06

II Xopt-X0 II=500 IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	2,66E-05	6,19E-12	2,57E-12
10^4	1,79E+01	1,83E+00	2,01E-10
10^6	2,07E+01	1,48E+00	1,71E-08
10^8	2,40E+01	5,56E+00	1,10E-05

Hypo1 zafixovanie kappa - rastom rozmeru rastie chyba **Kappa=10^2**

II Xopt-X0 II=5

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	7,39E-01	4,13E-01	2,91E-02
50	6,37E-02	1,72E-02	1,64E-06
100	4,82E-03	3,36E-05	3,32E-06
200	5,00E-04	4,31E-06	5,51E-06
300	2,11E-05	6,11E-06	9,11E-06

II Xopt-X0 II=10

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	1,58E+00	8,01E-01	6,85E-02
50	1,02E-01	4,46E-02	1,75E-06
100	7,90E-03	8,60E-05	3,73E-06
200	1,22E-04	4,30E-06	6,77E-06
300	5,72E-06	6,38E-06	8,24E-06

II Xopt-X0 II=50

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	8,29E+00	6,29E+00	8,91E-01
50	4,11E-01	2,46E-01	2,02E-06
100	7,88E-02	6,73E-04	3,33E-06
200	5,10E-04	4,09E-06	6,06E-06
300	5,96E-05	6,53E-06	8,41E-06

II Xopt-X0 II=100

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	1,53E+01	1,18E+01	3,71E+00
50	6,45E-01	3,02E-01	1,91E-06
100	4,67E-02	1,71E-03	3,48E-06
200	4,85E-02	4,70E-06	5,85E-06
300	5,87E-04	5,64E-06	8,34E-06

II Xopt-X0 II=200

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	3,45E+01	3,10E+01	2,03E+01
50	1,94E+00	1,34E+00	2,66E-05
100	2,82E-01	2,45E-03	3,18E-06
200	6,26E-04	4,13E-06	5,89E-06
300	4,13E-04	5,62E-06	7,10E-06

II Xopt-X0 II=500

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	7,34E+01	5,61E+01	6,41E+01
50	3,59E+00	2,38E+00	1,32E-02
100	4,03E-01	1,49E-02	3,54E-06
200	1,82E-02	4,57E-06	6,64E-06
300	7,17E-05	5,78E-06	7,66E-06

Hypo1 zafixovanie kappa - rastom rozmeru rastie chyba **Kappa=10^4**

II Xopt-X0 II=5

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	1,14E+00	8,94E-01	2,08E+00
50	6,73E-01	5,85E-01	4,33E-02
100	4,60E-01	3,64E-01	5,88E-05
200	2,43E-01	1,87E-01	7,85E-06
300	2,42E-01	1,78E-01	1,03E-05

II Xopt-X0 II=10

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	2,71E+00	2,18E+00	8,21E+00
50	1,29E+00	1,17E+00	6,18E-03
100	7,43E-01	6,51E-01	1,44E-05
200	6,67E-01	5,26E-01	8,70E-06
300	4,36E-01	3,13E-01	1,03E-05

II Xopt-X0 II=50

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	1,44E+01	1,34E+01	2,55E+01
50	5,54E+00	4,81E+00	1,60E+00
100	4,51E+00	3,46E+00	2,74E-03
200	3,14E+00	2,43E+00	8,11E-06
300	2,25E+00	1,52E+00	1,02E-05

II Xopt-X0 II=100

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	3,24E+01	3,08E+01	4,65E+01
50	1,15E+01	9,57E+00	1,30E+00
100	8,78E+00	7,49E+00	2,34E-02
200	7,42E+00	5,16E+00	1,48E-05
300	5,77E+00	4,01E+00	1,04E-05

II Xopt-X0 II=200

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	4,97E+01	4,72E+01	6,67E+01
50	2,33E+01	2,11E+01	4,19E+00
100	2,03E+01	1,80E+01	2,23E-01
200	1,16E+01	9,11E+00	2,09E-05
300	1,07E+01	8,23E+00	1,00E-05

II Xopt-X0 II=500

	IIXaprox-XoptII		
rozmer	bmpPSB	bpPSB	bBFGS
10	8,67E+01	4,41E+01	9,71E+01
50	6,29E+01	6,04E+01	6,39E+01
100	5,43E+01	4,32E+01	1,67E+01
200	3,97E+01	2,41E+01	8,27E-03
300	2,18E+01	1,36E+01	1,04E-05

Hypo1 zafixovanie kappa - rastom rozmeru rastie chyba **Kappa=10^6**

II Xopt-X0 II=5 — IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	9,22E-01	1,21E+00	1,91E+00
50	5,45E-01	5,73E-01	7,01E-01
100	4,26E-01	4,02E-01	2,64E-04
200	2,82E-01	2,50E-01	1,27E-05
300	2,91E-01	2,62E-01	1,40E-05

II Xopt-X0 II=10 — IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	2,22E+00	3,10E+00	3,45E+00
50	1,45E+00	1,41E+00	2,57E+00
100	8,04E-01	7,49E-01	1,48E-03
200	6,62E-01	6,07E-01	1,19E-05
300	6,30E-01	5,65E-01	1,34E-05

II Xopt-X0 II=50 — IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	1,73E+01	1,69E+01	2,61E+01
50	7,15E+00	6,58E+00	1,37E+01
100	5,62E+00	4,99E+00	3,15E-01
200	3,37E+00	3,02E+00	3,91E-05
300	2,88E+00	2,53E+00	1,32E-05

II Xopt-X0 II=100 — IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	1,78E+01	1,43E+01	1,42E+01
50	1,18E+01	1,11E+01	1,52E+01
100	1,02E+01	9,59E+00	8,30E+00
200	7,22E+00	5,93E+00	8,53E-03
300	4,77E+00	4,09E+00	1,51E-05

II Xopt-X0 II=200 — IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	2,17E+01	3,88E+01	6,51E+01
50	1,62E+01	2,37E+01	3,22E+01
100	1,92E+01	1,77E+01	7,26E+00
200	1,38E+01	1,16E+01	1,42E-03
300	1,21E+01	9,91E+00	2,23E-05

II Xopt-X0 II=500 — IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	7,35E+01	1,19E+02	8,64E+01
50	5,80E+01	5,28E+01	5,43E+01
100	3,16E+01	2,42E+01	3,64E+01
200	9,47E+00	1,79E+01	2,06E-01
300	6,68E+00	6,38E+00	2,67E-04

Hypo1 zafixovanie kappa - rastom rozmeru rastie chyba **Kappa=10^8**

II Xopt-X0 II=5 IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	1,09E+00	9,32E-01	3,00E+00
50	7,08E-01	6,85E-01	1,05E+00
100	3,88E-01	3,75E-01	1,16E-01
200	4,18E-01	3,73E-01	1,34E-01
300	2,76E-01	2,30E-01	2,00E-01

II Xopt-X0 II=10 IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	2,97E+00	2,62E+00	5,99E+00
50	1,40E+00	1,31E+00	2,94E+00
100	8,17E-01	7,22E-01	2,15E-01
200	6,21E-01	5,97E-01	1,81E-01
300	5,43E-01	4,83E-01	5,02E-01

II Xopt-X0 II=50 IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	1,49E+01	1,29E+01	1,74E+01
50	6,49E+00	5,98E+00	1,82E+01
100	4,42E+00	3,78E+00	4,97E+00
200	3,00E+00	2,62E+00	2,74E+00
300	1,96E+00	1,69E+00	8,67E+00

II Xopt-X0 II=100 IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	2,26E+01	2,09E+01	2,73E+01
50	1,52E+01	1,45E+01	2,48E+01
100	9,58E+00	8,12E+00	1,87E+01
200	7,81E+00	6,80E+00	1,23E+01
300	6,65E+00	5,86E+00	2,52E+01

II Xopt-X0 II=200 IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	7,32E+01	7,02E+01	8,02E+01
50	2,54E+01	2,37E+01	9,01E+01
100	1,99E+01	1,06E+01	4,01E+01
200	1,38E+01	1,26E+01	5,45E+00
300	1,03E+01	9,05E+00	3,42E+00

II Xopt-X0 II=500 IIXaprox-XoptII

rozmer	bmpPSB	bpPSB	bBFGS
10	9,75E+01	9,88E+01	9,77E+01
50	6,46E+01	5,99E+01	8,87E+01
100	3,93E+01	3,35E+01	3,60E+01
200	2,74E+01	2,41E+01	4,52E+01
300	2,28E+01	2,07E+01	7,70E+01

Hypo2 zafixovanie rozmeru- rastom kappa rastie chyba **Rozmer=10**

II Xopt-X0 II=5 — IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	7,39E-01	4,13E-01	2,91E-02
10^4	1,14E+00	8,94E-01	2,08E+00
10^6	9,22E-01	1,21E+00	1,91E+00
10^8	1,09E+00	9,32E-01	3,00E+00

II Xopt-X0 II=10 — IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	1,58E+00	8,01E-01	6,85E-02
10^4	2,71E+00	2,18E+00	8,21E+00
10^6	2,22E+00	3,10E+00	3,45E+00
10^8	2,97E+00	2,62E+00	5,99E+00

II Xopt-X0 II=50 — IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	8,29E+00	6,29E+00	8,91E-01
10^4	1,44E+01	1,34E+01	2,55E+01
10^6	1,73E+01	1,69E+01	2,61E+01
10^8	1,49E+01	1,29E+01	1,74E+01

II Xopt-X0 II=100 — IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	1,53E+01	1,18E+01	3,71E+00
10^4	3,24E+01	3,08E+01	4,65E+01
10^6	1,78E+01	1,43E+01	1,42E+01
10^8	2,26E+01	2,09E+01	2,73E+01

II Xopt-X0 II=200 — IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	3,45E+01	3,10E+01	2,03E+01
10^4	4,97E+01	4,72E+01	6,67E+01
10^6	2,17E+01	3,88E+01	6,51E+01
10^8	7,32E+01	7,02E+01	8,02E+01

II Xopt-X0 II=500 — IIXaprox-XoptII

kappa	mpPSB	pPSB	BFGS
10^2	7,34E+01	5,61E+01	6,41E+01
10^4	8,67E+01	4,41E+01	9,71E+01
10^6	7,35E+01	1,19E+02	8,64E+01
10^8	9,75E+01	9,88E+01	9,77E+01

Hypo2 zafixovanie rozmeru- rastom kappa rastie chyba **Rozmer=50**

II Xopt-X0 II=5

kappa	mpPSB	pPSB	BFGS
		IIXaprox-XoptII	
10^2	6,37E-02	1,72E-02	1,64E-06
10^4	6,73E-01	5,85E-01	4,33E-02
10^6	5,45E-01	5,73E-01	7,01E-01
10^8	7,08E-01	6,85E-01	1,05E+00

II Xopt-X0 II=10

kappa	mpPSB	pPSB	BFGS
		IIXaprox-XoptII	
10^2	1,02E-01	4,46E-02	1,75E-06
10^4	1,29E+00	1,17E+00	6,18E-03
10^6	1,45E+00	1,41E+00	2,57E+00
10^8	1,40E+00	1,31E+00	2,94E+00

II Xopt-X0 II=50

kappa	mpPSB	pPSB	BFGS
		IIXaprox-XoptII	
10^2	4,11E-01	2,46E-01	2,02E-06
10^4	5,54E+00	4,81E+00	1,60E+00
10^6	7,15E+00	6,58E+00	1,37E+01
10^8	6,49E+00	5,98E+00	1,82E+01

II Xopt-X0 II=100

kappa	mpPSB	pPSB	BFGS
		IIXaprox-XoptII	
10^2	6,45E-01	3,02E-01	1,91E-06
10^4	1,15E+01	9,57E+00	1,30E+00
10^6	1,18E+01	1,11E+01	1,52E+01
10^8	1,52E+01	1,45E+01	2,48E+01

II Xopt-X0 II=200

kappa	mpPSB	pPSB	BFGS
		IIXaprox-XoptII	
10^2	1,94E+00	1,34E+00	2,66E-05
10^4	2,33E+01	2,11E+01	4,19E+00
10^6	1,62E+01	2,37E+01	3,22E+01
10^8	2,54E+01	2,37E+01	9,01E+01

II Xopt-X0 II=500

kappa	mpPSB	pPSB	BFGS
		IIXaprox-XoptII	
10^2	3,59E+00	2,38E+00	1,32E-02
10^4	6,29E+01	6,04E+01	6,39E+01
10^6	5,80E+01	5,28E+01	5,43E+01
10^8	6,46E+01	5,99E+01	8,87E+01

Hypo2 zafixovanie rozmeru- rastom kappa rastie chyba **Rozmer=100**

II Xopt-X0 II=5

	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	4,82E-03	3,36E-05	3,32E-06
10^4	4,60E-01	3,64E-01	5,88E-05
10^6	4,26E-01	4,02E-01	2,64E-04
10^8	3,88E-01	3,75E-01	1,16E-01

II Xopt-X0 II=50

	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	7,88E-02	6,73E-04	3,33E-06
10^4	4,51E+00	3,46E+00	2,74E-03
10^6	5,62E+00	4,99E+00	3,15E-01
10^8	4,42E+00	3,78E+00	4,97E+00

II Xopt-X0 II=200

	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	2,82E-01	2,45E-03	3,18E-06
10^4	2,03E+01	1,80E+01	2,23E-01
10^6	1,92E+01	1,77E+01	7,26E+00
10^8	1,99E+01	1,06E+01	4,01E+01

II Xopt-X0 II=10

	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	7,90E-03	8,60E-05	3,73E-06
10^4	7,43E-01	6,51E-01	1,44E-05
10^6	8,04E-01	7,49E-01	1,48E-03
10^8	8,17E-01	7,22E-01	2,15E-01

II Xopt-X0 II=100

	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	4,67E-02	1,71E-03	3,48E-06
10^4	8,78E+00	7,49E+00	2,34E-02
10^6	1,02E+01	9,59E+00	8,30E+00
10^8	9,58E+00	8,12E+00	1,87E+01

II Xopt-X0 II=500

	IIXaprox-XoptII		
kappa	mpPSB	pPSB	BFGS
10^2	4,03E-01	1,49E-02	3,54E-06
10^4	5,43E+01	4,32E+01	1,67E+01
10^6	3,16E+01	2,42E+01	3,64E+01
10^8	3,93E+01	3,35E+01	3,60E+01

Hypo2 zafixovanie rozmeru- rastom kappa rastie chyba **Rozmer=300**

II Xopt-X0 II=5		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	2,11E-05	6,11E-06	9,11E-06
10^4	2,42E-01	1,78E-01	1,03E-05
10^6	2,91E-01	2,62E-01	1,40E-05
10^8	2,76E-01	2,30E-01	2,00E-01

II Xopt-X0 II=10		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	5,72E-06	6,38E-06	8,24E-06
10^4	4,36E-01	3,13E-01	1,03E-05
10^6	6,30E-01	5,65E-01	1,34E-05
10^8	5,43E-01	4,83E-01	5,02E-01

II Xopt-X0 II=50		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	5,96E-05	6,53E-06	8,41E-06
10^4	2,25E+00	1,52E+00	1,02E-05
10^6	2,88E+00	2,53E+00	1,32E-05
10^8	1,96E+00	1,69E+00	8,67E+00

II Xopt-X0 II=100		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	5,87E-04	5,64E-06	8,34E-06
10^4	5,77E+00	4,01E+00	1,04E-05
10^6	4,77E+00	4,09E+00	1,51E-05
10^8	6,65E+00	5,86E+00	2,52E+01

II Xopt-X0 II=200		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	4,13E-04	5,62E-06	7,10E-06
10^4	1,07E+01	8,23E+00	1,00E-05
10^6	1,21E+01	9,91E+00	2,23E-05
10^8	1,03E+01	9,05E+00	3,42E+00

II Xopt-X0 II=500		IIXaprox-XoptII	
kappa	mpPSB	pPSB	BFGS
10^2	7,17E-05	5,78E-06	7,66E-06
10^4	2,18E+01	1,36E+01	1,04E-05
10^6	6,68E+00	6,38E+00	2,67E-04
10^8	2,28E+01	2,07E+01	7,70E+01

Matlab zdrojový kód :

```
function [ presnost ] = mPSB4( G,h,xopt,x0 )
%Kvazinewtonovska metoda PSB
%   Minimalizacia kvadratickej funkcie so znamym riesenim Xopt
ro=norm(x0-xopt);
epsilon=ro/10;
ciara='\n============================================================
=============================';
fprintf('\n');
fprintf('\n Tabulka modifikovanych PSB iteracii ');
fprintf('\n');
fprintf(ciara);
fprintf('\n   k  eps.    |xk-xopt|    |p|      Fk-Fopt    |gradF|
Lambda      dTy       |Hk|');
fprintf(ciara);

Fopt=(xopt'*G*xopt)/2 + h'*xopt;
   n=length(h);

k=0;  %start z bodu x0
   x = x0;
   g = G*x+h;
   g0 = g;
k=1;  %Prva iteracia - Vypocet x1
   H = eye(n);
   s = -g;
   Lambda = -(g'*s)/(s'*G*s);
   p = Lambda*s;
   x = x+p;
   g = G*x+h;
   %cos_fi = (s'*g)/(norm(s)*norm(g));
   Fx = (x'*G*x)/2+h'*x;
      fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %s
%10.2e',...
          k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ' ,norm(H,'fro'));
k=2;   %Druha iteracia - Vypocet bodu x2
   y0 = g-g0;
   d = y0;
   r = p-y0;
   g0 = g;
   yTr = y0'*r;
   yTd = y0'*d;
   d0 = d/yTd;
   d1 = yTr*d0;
     H = H+d0*r'+(r-d1)*d0';
     s = -H*g;
     Lambda = -(g'*s)/(s'*G*s);
     p = Lambda*s;
     x = x+p;
     g = G*x+h;
     %cos_fi = (s'*g)/(norm(s)*norm(g));
     Fx = (x'*G*x)/2+h'*x;
   fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %11.2e
%10.2e',...
```

```
          k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,
yTd,norm(H,'fro'));

for k=3:(n);
    y = g-g0;
    r = p-H*y;
    beta = -(y0'*y)/(d'*y0);
    d = beta*d+y;
    yTr = y'*r;
    yTd = y'*d;
    d0 = d/yTd;
    d1 = yTr*d0;
    H = H+d0*r'+(r-d1)*d0';
    s = -H*g;
    Lambda = -(g'*s)/(s'*G*s);
    p = Lambda*s;
    x = x+p;
    g0 = g;
    g = G*x+h;
    Fx = (x'*G*x)/2 + h'*x;
    y0 = y;
    %cos_fi = (s'*g)/(norm(s)*norm(g));
    if norm(x-xopt) < epsilon
        fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e
%11.2e %10.2e',...
         k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,
yTd,norm(H,'fro'));
     epsilon=epsilon/10;
    end
     %tlac poslednej iteracie
        if k==n
            fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e
%10.2e %11.2e %10.2e',...
            k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,
yTd,norm(H,'fro'));
        end
    %tlac iteracie o jedno viac ako n
        if k==(n+1)
            fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e
%10.2e %11.2e %10.2e',...
            k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,
yTd,norm(H,'fro'));
        end
end;
presnost=norm(x - xopt);
fprintf(ciara);
fprintf('\n');
fprintf('    presnost = |xn-xopt| = %15.9f', presnost);
fprintf('\n');
end
```

```
function [ presnost ] = mpPSB( G,h,xopt,x0 )
%Kvazinewtonovska metoda PSB
%   Minimalizacia kvadratickej funkcie so znamym riesenim Xopt
ro=norm(x0-xopt);
epsilon=ro/10;
ciara='\n============================================================
==============================';
fprintf('\n');
fprintf('\n Tabulka modifikovanych pseudo PSB iteracii ');
fprintf('\n');
fprintf(ciara);
fprintf('\n   k  eps.    |xk-xopt|    |p|      Fk-Fopt    |gradF|
Lambda      dTy      |Hk|');
fprintf(ciara);

Fopt=(xopt'*G*xopt)/2 + h'*xopt;
   n=length(h);

k=0;  %start z bodu x0
   x = x0;
   g = G*x+h;
   g0 = g;
k=1;  %Prva iteracia - Vypocet x1
   H = eye(n);
   s = -g;
   Lambda = -(g'*s)/(s'*G*s);
   p = Lambda*s;
   x = x+p;
   g = G*x+h;
   %cos_fi = (s'*g)/(norm(s)*norm(g));
   Fx = (x'*G*x)/2+h'*x;
      fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %s
%10.2e',...
          k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A     ' ,norm(H,'fro'));
k=2;   %Druha iteracia - Vypocet bodu x2
   y0 = g-g0;
   d = y0;
   r = p-y0;
   g0 = g;
   rd=(r+y0);
     H = H+ ((rd*rd')/(y0'*rd))-((y0*y0')/(y0'*y0));
    s = -H*g;
     Lambda = -(g'*s)/(s'*G*s);
     p = Lambda*s;
     x = x+p;
     g = G*x+h;
     %cos_fi = (s'*g)/(norm(s)*norm(g));
     Fx = (x'*G*x)/2+h'*x;
   fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %s
%10.2e',...
          k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A     ',norm(H,'fro'));

for k=3:(n);
    y = g-g0;
    r = p-H*y;
    beta = -(y0'*y)/(d'*y0);
    d = beta*d+y;
    rd=(r+y);
    H=H+((rd*rd')/(y'*rd))-((y*y')/(y'*y));
```

```
    s = -H*g;
    Lambda = -(g'*s)/(s'*G*s);
    p = Lambda*s;
    x = x+p;
    g0 = g;
    g = G*x+h;
    Fx = (x'*G*x)/2 + h'*x;
    y0 = y;
    %cos_fi = (s'*g)/(norm(s)*norm(g));
    if norm(x-xopt) < epsilon
        fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %s
%10.2e',...
         k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ',norm(H,'fro'));
     epsilon=epsilon/10;
    end
     %tlac poslednej iteracie
        if k==n
            fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e
%10.2e %s %10.2e',...
            k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ',norm(H,'fro'));
        end
    %tlac iteracie o jedno viac ako n
        if k==(n+1)
            fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e
%10.2e %s %10.2e',...
            k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ',norm(H,'fro'));
        end
end;
presnost=norm(x - xopt);
fprintf(ciara);
fprintf('\n');
fprintf('    presnost = |xn-xopt| = %15.9f', presnost);
fprintf('\n');
end
```

```
function [ odchylka ] = BFGS( G,h,X_opt,X_0 )
%BFGS na riesenie kvadratickych funkcii
n=length(h);
H=eye(n);
X=X_0;
g=G*X_0+h;
lambda=-((g'*g)/(g'*G*g));
s=-H*g;
for k=1:n;
    X_n=X+lambda*s;
    p=X_n-X;
    g_n=G*X_n+h;

    %aktualizacia H
    y=g_n-g;

    A=(eye(n)-((y*p')/(y'*p)));
    z=((p*p')/(y'*p));

    H=A'*H*A+z     ;

    %presuny
    g=g_n;
    X=X_n;

    %krok a smer
    s=-H*g  ;      %smer'
    lambda=-((g'*s)/(s'*G*s))  ;  %krok
end
odchylka=norm(X_opt-X);
end
```

```
function [ odchylka ] = bBFGS( D,G,h,X_opt,X_0,ZR )
%BFGS na riesenie bikvadratickych funckii
n=length(h);
H=eye(n);
X=X_0;
g=((X'*D*X)*D+G)*X+h;
s=-H*g;
lambda=ZR2(D,G,h,X,s,ZR);
a=[lambda];
r=[norm(X_opt-X)];
for k=1:(n);
    X_n=X+lambda*s;
    p=X_n-X;
    g_n=((X_n'*D*X_n)*D+G)*X_n+h;

    %aktualizacia H
    y=g_n-g;

    A=(eye(n)-((y*p')/(y'*p)));
    z=((p*p')/(y'*p));

    H= A'*H*A+z     ;

    %presuny
    g=g_n;
    X=X_n;

    %krok a smer
    s=-H*g  ;       %smer'
    lambda=ZR2(D,G,h,X,s,ZR)  ;  %krok
    a=[a,lambda];
    r=[r,norm(X_opt-X)];
end
odchylka=norm(X_opt-X);
end
```

```
function [ presnost ] = bmPSB4( D,G,h,xopt,x0,ZR )
%Kvazinewtonovska metoda PSB
%   Minimalizacia bikvadratickej funkcie so znamym riesenim Xopt
ro=norm(x0-xopt);
epsilon=ro/10;
ciara='\n============================================================
==============================';
fprintf('\n');
fprintf('\n Tabulka modifikovanych PSB iteracii ');
fprintf('\n');
fprintf(ciara);
fprintf('\n   k  eps.    |xk-xopt|    |p|      Fk-Fopt    |gradF|
Lambda      dTy      |Hk|');
fprintf(ciara);

Fopt=(xopt'*G*xopt)/2 + h'*xopt;
   n=length(h);

k=0;  %start z bodu x0
   x = x0;
   g = ((x'*D*x)*D+G)*x+h;
   g0 = g;
k=1;  %Prva iteracia - Vypocet x1
   H = eye(n);
   s = -g;
   Lambda = ZR2(D,G,h,x,s,ZR)   ;
   p = Lambda*s;
   x = x+p;
   g = ((x'*D*x)*D+G)*x+h;
   %cos_fi = (s'*g)/(norm(s)*norm(g));
   Fx = ((x'*D*x)^2)/4+(x'*G*x)/2+h'*x;
      fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %s
%10.2e',...
          k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ' ,norm(H,'fro'));
k=2;  %Druha iteracia - Vypocet bodu x2
   y0 = g-g0;
   d = y0;
   r = p-y0;
   g0 = g;
   yTr = y0'*r;
   yTd = y0'*d;
   d0 = d/yTd;
   d1 = yTr*d0;
     H = H+d0*r'+(r-d1)*d0';
     s = -H*g;
     Lambda = ZR2(D,G,h,x,s,ZR)   ;
     p = Lambda*s;
     x = x+p;
     g = ((x'*D*x)*D+G)*x+h;
     %cos_fi = (s'*g)/(norm(s)*norm(g));
     Fx = ((x'*D*x)^2)/4+(x'*G*x)/2+h'*x;
   fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %11.2e
%10.2e',...
          k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,
yTd,norm(H,'fro'));

for k=3:(n+1);
    y = g-g0;
    r = p-H*y;
    beta = -(y0'*y)/(d'*y0);
```

```
    d = beta*d+y;
    yTr = y'*r;
    yTd = y'*d;
    d0 = d/yTd;
    d1 = yTr*d0;
    H = H+d0*r'+(r-d1)*d0';
    s = -H*g;
    Lambda = ZR2(D,G,h,x,s,ZR)  ;
    p = Lambda*s;
    x = x+p;
    g0 = g;
    g = ((x'*D*x)*D+G)*x+h;
    Fx = ((x'*D*x)^2)/4+(x'*G*x)/2+h'*x;
    y0 = y;
    %cos_fi = (s'*g)/(norm(s)*norm(g));
    if norm(x-xopt) < epsilon
        fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e
%11.2e %10.2e',...
         k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,
yTd,norm(H,'fro'));
     epsilon=epsilon/10;
    end
     %tlac poslednej iteracie
        if k==n
            fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e
%10.2e %11.2e %10.2e',...
            k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,
yTd,norm(H,'fro'));
        end
    %tlac iteracie o jedno viac ako n
        if k==(n+1)
            fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e
%10.2e %11.2e %10.2e',...
            k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,
yTd,norm(H,'fro'));
        end
end;
presnost=norm(x - xopt);
fprintf(ciara);
fprintf('\n');
fprintf('    presnost = |xn-xopt| = %15.9f', presnost);
fprintf('\n');
end
```

```
function [ presnost ] = bmpPSB( D,G,h,xopt,x0,ZR )
%Kvazinewtonovska metoda PSB
%   Minimalizacia bikvadratickej funkcie so znamym riesenim Xopt
ro=norm(x0-xopt);
epsilon=ro/10;
ciara='\n============================================================
==============================';
fprintf('\n');
fprintf('\n Tabulka modifikovanych pseudo PSB iteracii ');
fprintf('\n');
fprintf(ciara);
fprintf('\n   k  eps.    |xk-xopt|    |p|      Fk-Fopt    |gradF|
Lambda      dTy       |Hk|');
fprintf(ciara);

Fopt=(xopt'*G*xopt)/2 + h'*xopt;
   n=length(h);

k=0;  %start z bodu x0
   x = x0;
   g = ((x'*D*x)*D+G)*x+h;
   g0 = g;
k=1;  %Prva iteracia - Vypocet x1
   H = eye(n);
   s = -g;
   Lambda=ZR2(D,G,h,x,s,ZR)  ;
   p = Lambda*s;
   x = x+p;
   g = ((x'*D*x)*D+G)*x+h;
   %cos_fi = (s'*g)/(norm(s)*norm(g));
   Fx = ((x'*D*x)^2)/4+(x'*G*x)/2+h'*x;
      fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %s
%10.2e',...
          k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ' ,norm(H,'fro'));
k=2;  %Druha iteracia - Vypocet bodu x2
   y0 = g-g0;
   d = y0;
   r = p-y0;
   g0 = g;
  a = y0;

  H=H+ (((r+a)*(r+a)') /  (y0'*(r+a)))  -  (a*a') / (y0'*a) ;
    s = -H*g;
     Lambda =ZR2(D,G,h,x,s,ZR)  ;
     p = Lambda*s;
     x = x+p;
     g = ((x'*D*x)*D+G)*x+h;
     %cos_fi = (s'*g)/(norm(s)*norm(g));
     Fx = ((x'*D*x)^2)/4+(x'*G*x)/2+h'*x;
   fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %s
%10.2e',...
          k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ',norm(H,'fro'));

for k=3:(n);
    y = g-g0;
    r = p-H*y;
    beta = -(y0'*y)/(d'*y0);
    a = beta*a+y;
```

```
  H=H+ (((r+a)*(r+a)') /  (y'*(r+a)))  -  (a*a') / (y'*a) ;

    s = -H*g;
    Lambda =ZR2(D,G,h,x,s,ZR)  ;
    p = Lambda*s;
    x = x+p;
    g0 = g;
    g = ((x'*D*x)*D+G)*x+h;
    Fx = ((x'*D*x)^2)/4+(x'*G*x)/2+h'*x;
    y0 = y;
    %cos_fi = (s'*g)/(norm(s)*norm(g));
    if norm(x-xopt) < epsilon
        fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e  %10.2e %s
%10.2e',...
         k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ',norm(H,'fro'));
     epsilon=epsilon/10;
    end
     %tlac poslednej iteracie
        if k==n
            fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e
%10.2e %s %10.2e',...
            k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ',norm(H,'fro'));
        end
    %tlac iteracie o jedno viac ako n
        if k==(n+1)
            fprintf('\n %3.0f %1.0e %10.2e %10.2e %10.2e %11.2e
%10.2e %s %10.2e',...
            k,epsilon,norm(x-xopt), norm(p),Fx-Fopt,norm(g),Lambda,'
N/A    ',norm(H,'fro'));
        end
end;
presnost=norm(x - xopt);
fprintf(ciara);
fprintf('\n');
fprintf('    presnost = |xn-xopt| = %15.9f', presnost);
fprintf('\n');
end
```

```
function[aprox]=ZR2(D,G,h,x,s,ZR)

%hladanie intervalu
a=0;
b=1;

fi0=ffi(D,G,h,x,s,a);
fi1=ffi(D,G,h,x,s,b);

while fi1<fi0
    b=b+0.5;
    fi1=ffi(D,G,h,x,s,b);
end

%konstanta
k1=0.5*(sqrt(5)-1);
k2=0.5*(3-sqrt(5));

%prvz krok
c1=a+k2*(b-a);
c2=a+k1*(b-a);
fi1=ffi(D,G,h,x,s,c1);
fi2=ffi(D,G,h,x,s,c2);
%algorytmus
for i=1:ZR
    if fi1<fi2

        b=c2;
        c2=c1;
        c1=a+k2*(b-a);
    else
        a=c1;
        c1=c2;
        c2=a+k1*(b-a);
    end
    fi1=ffi(D,G,h,x,s,c1);
    fi2=ffi(D,G,h,x,s,c2);
end
aprox=0.5*(a+b);
```

```
function [ hodnota ] = ffi( De,Ge,ha,x,s,lambda )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here
z=x+lambda*s;
hodnota=0.25*((z'*De*z)^2)+0.5*(z'*Ge*z)+ha'*z;

end
```

```
function [ G,h,X_opt,X_nul ] = Generator( n,kappa,ro )
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here
%Generujeme G
D=diag(1:((10^kappa-1)/(n-1)):10^kappa);
A=100*rand(n,n);
[Q,R]=qr(A);
G=Q*D*(Q');

%Generujeme X_opt
X_opt=100*rand(n,1);

%Doratame h
h=-G*X_opt;

%Generujeme X_nul
y=100*rand(n,1);
X_nul= X_opt+ro*((y-X_opt)/(norm(y-X_opt)));


end
```