

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

DIPLOMOVÁ PRÁCE

Nehladká Newtonova metoda a její použití



Vedoucí bakalářské práce:
RNDr. Horymír Netuka, Ph.D.
Rok odevzdání: 2011

Vypracovala:
Bc. Ivana Sýkorová
MPM, II. ročník

Prohlášení

Prohlašuji, že jsem diplomovou práci zpracovala samostatně pod vedením RNDr. Horymíra Netuky, Ph.D. s použitím uvedené literatury.

V Olomouci dne 30. března 2011

Poděkování

Chtěla bych na tomto místě poděkovat především svému vedoucímu diplomové práce RNDr. Horymíru Netukovi, Ph.D. za vstřícnost, podporu a cenné rady při psaní této práce. Také bych chtěla poděkovat prof. RNDr. Jaroslavu Haslingerovi, DrSc. za ochotu, trpělivost a podnětné připomínky. Nemalý dík patří i mé rodině a přátelům za podporu během studia.

Obsah

Úvod	4
1 Základní pojmy	6
2 Popis problému a princip řešení	9
3 Clarkovo zobecnění jakobiánu	14
4 Princip nehladké Newtonovy metody	26
5 Aplikace na funkce po částech hladké	39
6 Numerické výpočty	44
6.1 Formulace úlohy	44
6.2 Diskretizace úlohy	45
6.3 Počítačový program	51
6.4 Příklady	64
Závěr	76

Úvod

Tato diplomová práce si klade za úkol seznámit čtenáře s relativně novou metodou z oblasti nediferencovatelné optimalizace, která se začala úspěšně používat v různých aplikacích. Cílem práce je nastudovat teorii, a zaměřit se zejména na princip nehladké Newtonovy metody a na otázky konvergence, dále vypracovat počítačovou realizaci a provést numerické experimenty a výpočty zadaných příkladů.

Do první kapitoly jsou zařazeny pojmy, se kterými se čtenář setkává v průběhu celého textu. Připomenutím těch známějších jsou vytvořeny základy, na nichž jsou vystavěny pozdější výsledky. Termíny méně známé jsou zase vysvětleny na příkladu.

V kapitole druhé je zaveden pojem variační nerovnice a provedena diskuze o ekvivalentních vyjádřeních, alternativních formulacích, vztazích mezi nimi, a jejich interpretaci. V závěru je stanovena rovnice, kterou se budeme nadále zabývat, a vymezeny metody, jaké pro její řešení budeme konstruovat.

Kapitola třetí představuje nástroje pro zacházení s nehladkými funkcemi, jež jsou obdobou těch pro funkce hladké. Zahrnuty jsou výhody i nevýhody nově uvedených pojmů, početní pravidla, i obdoby vět známých pro hladké funkce. Celá tato část je doprovázena četnými ilustračními příklady.

Ve čtvrté kapitole jsou v úvodu analyzovány vlastnosti klasické Newtonovy metody a následně formalizovány pro použití na nehladké funkce. Zásadní je sestavení dvou algoritmů, diskuze nad jejich rozdíly a jednotlivými kroky a uvedení konvergenčních výsledků.

Kapitola následující se věnuje speciální třídě funkcí, důležitému příkladu a úpravě předcházejících algoritmů, jež budou využity v závěru práce. Opět jsou doplněny i odpovídající konvergenční věty.

V poslední kapitole je formulována konkrétní úloha, diskretizována a převedena do podoby, na níž lze aplikovat Newtonův algoritmus. Je zde popsán počítačový program pro její řešení, a použit na ukázkové a zadané příklady.

Vlastní přínos představuje v první kapitole zvážení, které pojmy zahrnout, a

jejich dohledání. Také jsem našla a vypracovala vhodný příklad. V druhé kapitole jsem provedla srovnání s variační nerovnicí v nekonečné dimenzi. V kapitole třetí jsem u příkladů 3.1, 3.5 zachovala postup, ale změnila zadání, řešení a výsledky. V páté kapitole jsem rozšířila a doplnila vysvětlujícími poznámkami příklad 5.1. Stěžejní částí je pak kapitola poslední, počínaje formulací úlohy, přes několikanásobné přeformulování, až k vytvoření a odladění programu, vymyšlení příkladů, nakreslení schémat a dokumentaci výsledků.

1 Základní pojmy

Připomeňme si terminologii z různých oblastí, na jejích základech budeme v průběhu celého textu stavět.

Definice 1.1. *Nechť $f : \mathcal{D} \rightarrow \mathbb{R}^n$, $\mathcal{D} \subseteq \mathbb{R}^n$ je otevřená množina, $x \in \mathcal{D}$, $\alpha \in \mathbb{R}^n$. Existuje-li limita*

$$\lim_{t \downarrow 0} \frac{f(x + t\alpha) - f(x)}{t} = f'_\alpha(x),$$

nazýváme ji směrovou derivací funkce f v bodě x podle vektoru α . Jestliže $\|\alpha\| = 1$, hovoříme o směrové derivaci ve směru α .

Směrové derivace funkce f v bodě x ve směru vektorů \mathbf{e}_i , $i = 1, \dots, n$, kanonické báze \mathbb{R}^n nazveme parciálními derivacemi funkce f v bodě x podle i -té proměnné x_i . Značíme $\frac{\partial f}{\partial x_i}(x)$, $i = 1, \dots, n$.

Definice 1.2. *Nechť $f : \mathcal{D} \rightarrow \mathbb{R}^n$, $\mathcal{D} \subseteq \mathbb{R}^n$ je otevřená množina, $x \in \mathcal{D}$, $\alpha \in \mathbb{R}^n$. Říkáme, že funkce f je F -diferencovatelná (Fréchetovsky) v bodě x , existuje-li lineární forma $df(x, \cdot)$ na \mathbb{R}^n taková, že platí limitní podmínka*

$$\lim_{\alpha \rightarrow 0} \frac{f(x + \alpha) - f(x) - df(x, \alpha)}{\|\alpha\|} = 0.$$

Říkáme, že funkce f je F -diferencovatelná, je-li F -diferencovatelná v každém bodě $x \in \mathcal{D}$. Má-li spojité parciální derivace v každém bodě $x \in \mathcal{D}$, říkáme, že je třídy C^1 na množině \mathcal{D} . Značíme $f \in C^1(\mathcal{D})$.

Definice 1.3. *Uvažujme funkci $G = (g_1, \dots, g_m) : \mathcal{D} \rightarrow \mathbb{R}^m$, kde $\mathcal{D} \subseteq \mathbb{R}^n$ je otevřená množina, a dále $x \in \mathcal{D}$. Nechť funkce $g_i : \mathcal{D} \rightarrow \mathbb{R}$ mají všechny parciální derivace v bodě x . Potom matici*

$$JG(x) = \left(\frac{\partial g_i}{\partial x_j}(x) \right)_{i,j=1}^{m,n}$$

nazveme jakobiánem funkce G .

Definice 1.4. Funkce $f : \mathcal{D} \rightarrow \mathbb{R}^n$, kde $\mathcal{D} \subseteq \mathbb{R}^n$, se nazývá lipschitzovská, jestliže existuje reálná konstanta $L > 0$ taková, že pro každé $x, y \in \mathcal{D}$ platí

$$\|f(x) - f(y)\| \leq L\|x - y\|.$$

Funkce f se nazývá lokálně lipschitzovská, jestliže pro každý bod $x \in \mathcal{D}$ existuje jeho okolí, na němž je f lipschitzovská.

Definice 1.5. Zobrazení $\Phi : S \rightarrow T$, kde S a T jsou podmnožiny \mathbb{R}^n , se nazývá homeomorfismus z S na T , jestliže Φ je spojitý a bijektivní a jestliže inverzní zobrazení $\Phi^{-1} : T \rightarrow S$ je také spojitý.

Definice 1.6. Funkce $\Phi : \mathcal{D} \rightarrow \mathbb{R}^n$, kde \mathcal{D} je otevřená podmnožina \mathbb{R}^n , se nazývá

- (a) lokální homeomorfismus v bodě $x \in \mathcal{D}$, jestliže existuje otevřené okolí $\mathcal{N} \subseteq \mathcal{D}$ bodu x tak, že restrikce $\Phi|_{\mathcal{N}} : \mathcal{N} \rightarrow \Phi(\mathcal{N})$ je homeomorfismus; zobrazení $(\Phi|_{\mathcal{N}})^{-1}$ pak nazýváme lokální inverze Φ ;
- (b) lokálně lipschitzovský homeomorfismus v bodě $x \in \mathcal{D}$, jestliže existuje otevřené okolí $\mathcal{N} \subseteq \mathcal{D}$ bodu x tak, že restrikce $\Phi|_{\mathcal{N}} : \mathcal{N} \rightarrow \Phi(\mathcal{N})$ je homeomorfismus a obě zobrazení $\Phi|_{\mathcal{N}}$ a $(\Phi|_{\mathcal{N}})^{-1}$ jsou lipschitzovské na svých definičních oborech;
- (c) lipschitzovský homeomorfismus, jestliže $\mathcal{D} = \mathbb{R}^n$, Φ je homeomorfismus zobrazující \mathbb{R}^n na sebe a Φ i Φ^{-1} jsou lipschitzovské na \mathbb{R}^n .

Definice 1.7. Funkce $\Phi : \mathcal{D} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$ definovaná na otevřené množině \mathcal{D} se nazývá B-diferencovatelná (Bouligand-diferencovatelná) v bodě $x \in \mathcal{D}$, jestliže Φ je lipschitzovská v okolí x a má v x všechny směrové derivace. Jestliže je Φ B-diferencovatelná v x , nazýváme směrovou derivaci $\Phi'(x; d)$ B-derivací Φ v x podél d . B-derivace $\Phi'(x; \cdot)$ je silná, pokud chybová funkce

$$e(y) \equiv \Phi(y) - \Phi(x) - \Phi'(x; y - x)$$

splňuje

$$\lim_{\substack{y^1 \neq y^2 \\ (y^1, y^2) \rightarrow (x, x)}} \frac{e(y^1) - e(y^2)}{\|y^1 - y^2\|} = 0.$$

Pak říkáme, že Φ je silně B-diferencovatelná v x . Říkáme, že Φ je B-diferencovatelná blízko x , jestliže je B-diferencovatelná v každém bodě v jistém okolí x .

Ve skutečnosti je B-diferencovatelnost oslabením klasické myšlenky F-diferencovatelnosti, což ukazuje následující příklad.

Příklad 1.1. Uvažujme funkci

$$f(x) = \begin{cases} x + x^2, & \text{jestliže } x \geq 0, \\ -x - x^2, & \text{jestliže } x < 0. \end{cases}$$

Tato funkce je lokálně lipschizovská v bodě 0 a má zde všechny směrové derivace, je tedy B-diferencovatelná. Na druhou stranu zde ale není F-diferencovatelná. Ukažme to tím, že do podmínky z definice 1.1 dosadíme funkci $f(x)$, bod $x = 0$ a α kladné, klesající k nule, a následně $-\alpha$. Po dosazení máme

$$0 = \lim_{\alpha \rightarrow 0} \frac{\alpha + \alpha^2 - df(0, \alpha)}{|\alpha|} = \lim_{\alpha \rightarrow 0} \left(1 + \alpha - \frac{df(0, \alpha)}{\alpha} \right) = 1 - \lim_{\alpha \rightarrow 0} \frac{df(0, \alpha)}{\alpha},$$
$$0 = \lim_{\alpha \rightarrow 0} \frac{\alpha - \alpha^2 - df(0, -\alpha)}{|-\alpha|} = \lim_{\alpha \rightarrow 0} \left(1 - \alpha + \frac{df(0, \alpha)}{\alpha} \right) = 1 + \lim_{\alpha \rightarrow 0} \frac{df(0, \alpha)}{\alpha}.$$

Porovnáním členů na pravých stranách dostáváme

$$-1 = 1,$$

tedy požadovaná podmínka není nikdy splněna, a funkce $f(x)$ vskutku není v 0 F-diferencovatelná.

2 Popis problému a princip řešení

V této kapitole se seznámíme s variačními nerovnicemi a úlohami komplementarity, které představují hlavní motivaci této práce. Také si nastíníme metody jejich řešení, a v kapitolách následujících se jimi budeme zabývat.

Nejjednodušším příkladem variační nerovnice je úloha vyřešit systém nelineárních rovnic. Jak brzy uvidíme, tento problém můžeme opravdu považovat za variační nerovnici bez omezujících podmínek. Nyní definujme variační nerovnici v obecném tvaru.

Definice 2.1. *Nechť je dána podmnožina K euklidovského n -dimenzionálního prostoru \mathbb{R}^n a zobrazení $F : K \rightarrow \mathbb{R}^n$. Pak variační nerovnicí, označovanou $VN(K, F)$, rozumíme úlohu najít vektor $x \in K$ takový, že*

$$(y - x)^T F(x) \geq 0, \quad \forall y \in K. \quad (2.1)$$

Množinu řešení této úlohy značíme $\mathcal{S}(K, F)$.

Poznámka 2.1. *V této práci se budeme zajímat pouze o situace, kdy je množina K uzavřená a funkce F spojitá. Spojitostí F budeme nadále rozumět spojitost F na otevřené množině obsahující K . Pokud bude vyžadována, stejnou úvahu provedeme pro diferencovatelnost F .*

Poznámka 2.2. *Pojem variační nerovnice známe také z variačního počtu, kde je její obecný tvar definován následovně. Je-li dán Banachův prostor E , podmnožina $K \in E$, a funkcionál $F : K \rightarrow E^*$ zobrazující K do duálu E^* , pak variační nerovnicí rozumíme úlohu vyřešit vzhledem k proměnné x náležející do K následující nerovnici:*

$$\langle F(x), y - x \rangle \geq 0, \quad \forall y \in K,$$

kde $\langle \cdot, \cdot \rangle : E^ \times E \rightarrow \mathbb{R}$ je dualita. Takto definovanou variační nerovnici můžeme považovat za zobecnění definice 2.1 do prostorů s nekonečnou dimenzí.*

Podívejme se blíže na definici 2.1. Dle známých vlastností skalárního součinu v \mathbb{R}^n je jeho výsledek kladný, pokud jsou oba vektory nenulové a svírají ostrý

úhel, a naopak záporný, pokud svírají úhel tupý. Z této úvahy plyne následující geometrická interpretace variační nerovnice.

Věta 2.1. *Bod x z množiny K je řešení $VN(K, F)$ tehdy a pouze tehdy, když $F(x)$ netvoří tupý úhel s každým vektorem tvaru $y - x$, pro všechna $y \in K$.*

Toto pozorování můžeme formalizovat užitím pojmu normálního kužele.

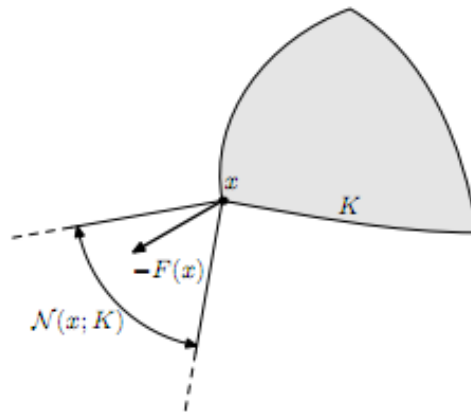
Definice 2.2. *Je-li dána množina $K \subseteq \mathbb{R}^n$ a vektor x' do ní náležející, definujeme normální kužel ke K v x' jako následující množinu:*

$$N(x'; K) \equiv \{d \in \mathbb{R}^n : d^T(y - x') \leq 0, \forall y \in K\}.$$

Vektory v této množině nazýváme normální vektory k množině K v x' .

Poznámka 2.3. *Nerovnost (2.1) tedy říká, že vektor $x \in K$ řeší $VN(K, F)$ tehdy a jen tehdy, když $-F(x)$ je normální vektor ke K v x , nebo ekvivalentně*

$$0 \in F(x) + N(x; K).$$



Obrázek 1: Řešení VN a normální kužel

Obrázek 1 znázorňuje tento úhel pohledu. Jak bylo zmíněno na začátku této kapitoly, nejjednodušší případ VN je úloha vyřešit soustavu nelineárních rovnic, která odpovídá tomu, že množina K je rovna celému prostoru \mathbb{R}^n .

Věta 2.2. *Jestliže $K = \mathbb{R}^n$, vektor x náleží do $\mathcal{S}(K, F)$ tehdy a jen tehdy, když x je nulový bod zobrazení F , to jest $F(x) = 0$.*

Důkaz: Máme ukázat, že $\mathcal{S}(\mathbb{R}^n, F) = F^{-1}(0)$. Všimněme si, že pro libovolnou množinu K , jestliže $x \in K$ a $F(x) = 0$, pak zřejmě $x \in \mathcal{S}(K, F)$. Proto $F^{-1}(0) \cap K$ je vždy podmnožina $\mathcal{S}(K, F)$.

K potvrzení opačné inkluze, kdy $K = \mathbb{R}^n$, poznamenejme, že

$$x \in \mathcal{S}(\mathbb{R}^n, F) \Rightarrow F(x)^T d \geq 0 \quad \forall d \in \mathbb{R}^n.$$

Vezmeme-li d jako $-F(x)$, vidíme, že $F(x) = 0$. Proto $\mathcal{S}(\mathbb{R}^n, F) \subseteq F^{-1}(0)$, tedy rovnost platí. \square

Výše uvedený argument platí obecněji pro řešení VN(K, F), které náleží do vnitřku $K \subset \mathbb{R}^n$.

Věta 2.3. *Jestliže x je řešení VN(K, F) a x náleží do vnitřku K , pak $F(x) = 0$.*

Důkaz: Pokud $x \in \text{int } K$, existuje skalár $\tau > 0$ dostatečně malý tak, že vektor $y \equiv x - \tau F(x)$ náleží do K . Dosadíme-li tento vektor do (2.1), vidíme, že $-F(x)^T F(x) \geq 0$, což implikuje $F(x) = 0$. \square

Jestliže K je kužel, to jest platí $x \in K \Rightarrow \tau x \in K$ pro všechny skaláry $\tau \geq 0$, VN připouští ekvivalentní tvar známý jako úloha komplementarity.

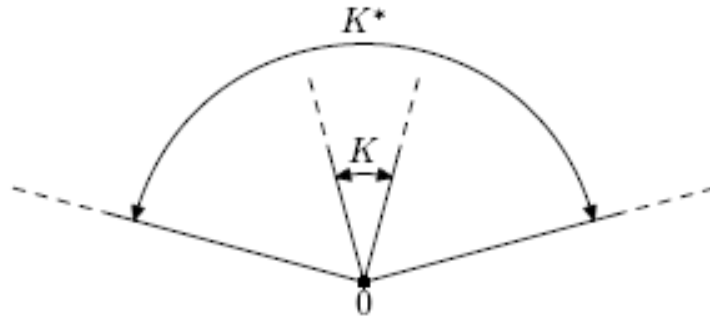
Definice 2.3. *Je-li dán kužel K a zobrazení $F : K \rightarrow \mathbb{R}^n$, úloha komplementarity, označovaná jako $UK(K, F)$, znamená úlohu najít vektor $x \in \mathbb{R}^n$ vyhovující následující podmínce:*

$$K \ni x \perp F(x) \in K^*,$$

kde \perp značí „perpendikulár“ a K^* je duální kužel ke K definovaný jako

$$K^* \equiv \{d \in \mathbb{R}^n : v^T d \geq 0 \quad \forall v \in K\},$$

to jest K^* sestává ze všech vektorů, které tvoří ostrý nebo pravý úhel s každým vektorem $z \in K$.



Obrázek 2: Kužel a kužel k němu duální

Poznámka 2.4. *Napíšeme-li notaci \perp explicitně, získáme podmínku ve tvaru nehladké rovnice:*

$$x \in K, \quad F(x) \in K^*, \quad x^T F(x) = 0.$$

V tuto chvíli jsme dosáhli kýžené formulace ve tvaru nehladké rovnice a v dalším se budeme zabývat především tímto pojmem. Popíšme ještě přesnou souvislost mezi $VN(K, F)$ a $UK(K, F)$, kde K je kužel.

Věta 2.4. *Nechť K je kužel v \mathbb{R}^n . Vektor x řeší $VN(K, F)$ tehdy a jen tehdy, když x řeší $UK(K, F)$.*

Důkaz: Viz [2], str. 4.

Poznámka 2.5. *Co se týče označení $UK(K, F)$, když je zkratka UK vztáhnuta ke dvojici (K, F) , rozumí se, že K je kužel.*

Nyní už tedy známe vztah mezi variačními nerovnicemi a nehladkými rovnicemi, na jejichž řešení se zaměříme. Uvažujme soustavu nehladkých rovnic ve tvaru

$$G(x) = 0, \tag{2.2}$$

kde $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ je lokálně lipschitzovská na otevřené množině Ω . Konkrétně se budeme soustředit na numerické metody newtonovského typu pro řešení takovýchto rovnic, a jejich lokální konvergenci.

Lokální konvergence algoritmus znamená, že po počátečním bodu požadujeme, aby náležel do vhodně zvoleného okolí řešení za účelem zajištění konvergence algoritmu. Naproti tomu je algoritmus globálně konvergentní, pokud tento požadavek na počáteční bod můžeme vypustit.

Je vhodné zvážit i úzce související úlohu nalézt řešení (2.2), které náleží do dané množiny X :

$$G(x) = 0, \quad x \in X, \quad (2.3)$$

kde $X \subseteq \Omega$ je uzavřená množina. Rovnici (2.3) nazýváme „rovnice s omezeními“. Pokud X odpovídá celému \mathbb{R}^n , (2.3) se redukuje na (2.2). Existují nejméně dva důvody pro zavedení množiny X . První z nich je, že pro variační nerovnici s množinou K a zobrazením F nemusí být F definováno vně otevřené množiny obsahující K . Tento znak pak nese také přeformulování variační nerovnice na úlohu komplementarity, neboli na nehladkou rovnici. Druhým důvodem pak je, že řešení původní úlohy i jakéhokoliv jejího přeformulování musí náležet do množiny K . Proto je smysluplné pokusit se zabudovat tuto informaci do algoritmu položením $X = K$.

Klasické metody obsahují mnoho myšlenek, které se pokusíme uplatnit, ovšem kvůli nediferencovatelnosti G je potřeba nejen tyto metody přepracovat, ale také představit nové nástroje.

3 Clarkovo zobecnění jakobiánu

V této kapitole uvedeme rozšíření některých klasických výsledků analýzy hladkých reálných funkcí na funkce nehladké. Hlavním nástrojem pro studium nehladkých funkcí je směrová derivace. Nicméně pokud bychom při navrhování algoritmu lpěli výhradně na tomto konceptu, můžeme narazit na mnoho obtíží. Proto pro lokálně lipschitzovské funkce představíme Clarkovo zobecnění pojmu gradientu a jakobiánu a odpovídající početní pravidla. Také se stručně dotkneme některých komplexnějších témat, jako jsou věta o střední hodnotě a věta o implicitní funkci. Výsledky v této části tvoří základ, na kterém můžeme vybudovat efektivní metody.

Definice 3.1. *Nechť $G : \Omega \rightarrow \mathbb{R}^m$, kde $\Omega \subseteq \mathbb{R}^n$ je otevřená, je lokálně lipschitzovská pro každý vektor $x \in \Omega$. Definujme limitní jakobián G v bodě \bar{x} jako*

$$\text{Jac } G(\bar{x}) = \partial_B G(\bar{x}) \equiv \left\{ H \in \mathbb{R}^{m \times n} : H = \lim_{k \rightarrow \infty} JG(x^k), \right. \\ \left. \text{kde posloupnost } \{x^k\} \rightarrow \bar{x}, x^k \notin N_G \right\},$$

kde jsme označili N_G množinu míry nula takových bodů, ve kterých G není F -diferencovatelná.

Poznámka 3.1. *Podle Rademacherovy věty existuje alespoň jedna posloupnost $\{x^k\}$ F -diferencovatelných bodů, která konverguje k \bar{x} . Jelikož jsou matice jakobiánu G v F -diferencovatelných bodech blízko \bar{x} omezené v normě lokální lipschitzovskou konstantou funkce G v bodě \bar{x} , plyne z toho, že všechny posloupnosti $\{JG(x^k)\}$, specifikované v definici 3.1, musí mít alespoň jeden hromadný bod. Proto je $\text{Jac}(G, \bar{x})$ neprázdná množina. Obecně pro nediferencovatelné funkce obsahuje $\text{Jac}(G, \bar{x})$ více matic.*

Přestože lze tento koncept úspěšně využít při studiu funkcí po částech hladkých, nemůžeme popřít fakt, že obecně je limitní jakobián obtížný objekt ke studiu. Pro obecnou nehladkou funkci není snadný ani výpočet, ani práce s limitním

jakobiánem. Abychom se alespoň částečně vyhnuli obtížím, představíme následující definici.

Definice 3.2. *Nechť $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$, kde Ω je otevřená, je lokálně lipschitzovská pro každý vektor $\bar{x} \in \Omega$. Clarkův zobecněný jakobián G v \bar{x} je definován takto:*

$$\partial G(\bar{x}) \equiv \text{conv Jac } G(\bar{x}). \quad (3.1)$$

Pro $m = 1$, tedy když G je reálná funkce $g : \mathbb{R}^n \rightarrow \mathbb{R}$, se $\partial g(x)$ nazývá zobecněný gradient g v \bar{x} . V souladu se značením gradientu hladké funkce jsou složky $\partial g(x)$ sloupcové vektory.

Často hovoříme o Clarkově zobecněném jakobiánu jednoduše jako o zobecněném jakobiánu G . Definici můžeme ilustrovat na jednoduché funkci $|x|$. Tato funkce je lipschitzovská s konstantou $L = 1$, a spojitě diferencovatelná všude kromě počátku. Je snadné ověřit, že v tomto bodě platí $\text{Jac } |0| = \{-1, 1\}$, a zobecněný gradient je pak jednoduše $\partial|0| = [-1, 1]$.

Poznámka 3.2. *Obecně může být výpočet zobecněného jakobiánu zjednodušen faktem, že zobecněný jakobián je „slepý“ k množinám míry nula ve smyslu, že jestliže N_0 je libovolná množina míry nula v \mathbb{R}^n , pak*

$$\partial G(\bar{x}) \equiv \text{conv} \{H : H = \lim_{k \rightarrow \infty} JG(x^k), \{x^k\} \rightarrow \bar{x}, x^k \notin N_G \cup N_0\}.$$

Tohoto výsledku můžeme použít, abychom ilustrovali výpočet zobecněného jakobiánu poněkud komplikovanější funkce, než je absolutní hodnota z x .

Příklad 3.1. *Mějme funkci ve tvaru:*

$$G(x, y) \equiv \begin{pmatrix} -2x + |y| \\ \min(x, y) \end{pmatrix}.$$

Abychom dostali $\partial G(0,0)$, uvažujme čtyři uzavřené oblasti

$$R_1 \equiv \{(x, y) : y \geq x, x \geq 0\},$$

$$R_2 \equiv \{(x, y) : y \geq x, x \leq 0\},$$

$$R_3 \equiv \{(x, y) : y \leq x, x \leq 0\},$$

$$R_4 \equiv \{(x, y) : y \leq x, x \geq 0\}.$$

Je zřejmé, že sjednocení těchto čtyř oblastí dává celý prostor a že sjednocení B jejich hranic je množina míry nula. Všimněte si, že množina B obsahuje všechny body, pro které je G nediferencovatelná, ale také body, ve kterých je diferencovatelná spojitě (např. kladná osa x). Na základě poznámky před tímto příkladem je pak snadné nahlédnout, že zobecněný jakobián G v $(0,0)$ je dán jako konvexní obal množiny limit jakobiánů G vypočítaných pro posloupnosti konvergující k nule zevnitř každé ze čtyř oblastí R_i :

$$\partial G(0,0) = \text{conv} \left(\left[\begin{array}{cc} -2 & 0 \\ 1 & 0 \end{array} \right], \left[\begin{array}{cc} -2 & 0 \\ 0 & 1 \end{array} \right] \right),$$

což potvrzuje výraz (3.1) z definice 3.2.

Jiná zajímavá funkce, pro niž můžeme na základě definice snadno vypočítat zobecněný gradient, je euklidovská norma.

Příklad 3.2. Necht' $g(x) = \|x\|_2$ je euklidovská norma v \mathbb{R}^n . Tato funkce je spojitě diferencovatelná všude kromě počátku. Jestliže $x \neq 0$, snadno dostaneme

$$\nabla \|x\|_2 = \frac{x}{\|x\|_2},$$

takže norma gradientu funkce v tomto tvaru je rovna jedné pro každý nenulový vektor. Z tohoto výpočtu je zřejmé, že $\text{Jac } g(0)$ sestává ze všech vektorů s euklidovskou normou rovnou jedné. Zobecněný gradient v počátku je konvexní obal této množiny a proto dostáváme $\partial \|0\| = \overline{B(0,1)}$. Povšimněme si, že $\text{Jac } g(0)$ je kompaktní, ale nekonvexní množina.

Z definice zobecněného Jakobiánu je zřejmé, že je-li reálná funkce g spojitě diferencovatelná v x , pak $\partial g(x)$ je roven jednobodové množině $\{\nabla g(x)\}$. Také lze ukázat, že jestliže je g lipschitzovská a gâteauxovsky diferencovatelná v x , pak $\nabla g(x) \in \partial g(x)$, ale zobecněný gradient je roven jednobodové množině $\{\nabla g(x)\}$ tehdy a jen tehdy, když má g F-derivaci v x . Projevují se zde nevýhody zobecněného jakobiánu. Můžeme říci, že samotný termín zobecněného jakobiánu je nepřesný, protože zobecněný jakobián se nemusí nutně zredukovat na obvyklý jakobián, jestliže je funkce gâteauxovsky diferencovatelná v bodě. V mnoha situacích se může zobecněný jakobián ukázat jako „příliš velká množina“. Z tohoto důvodu často uvažujeme speciální třídy lokálně lipschitzovských funkcí, pro které je Clarkův zobecněný jakobián vhodným nástrojem.

Poznámka 3.3. *Zobecněný jakobián může být vnímán jako zobrazení z Ω do podmnožin $\mathbb{R}^{n \times m}$:*

$$\partial G : x \in \Omega \mapsto \partial G(x) \subset \mathbb{R}^{n \times m}.$$

Níže jsou uvedeny některé základní vlastnosti tohoto zobrazení v bodě x za předpokladu lokální lipschitzovskosti.

Věta 3.1. *Nechť je dána funkce $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$, kde G je lokálně lipschitzovská na otevřené množině Ω . Následující výroky platí pro libovolné $x \in \Omega$:*

- (a) $\partial G(x)$ je neprázdná, konvexní a kompaktní;
- (b) zobrazení ∂G je shora polospojité v x , tedy pro každé $\varepsilon > 0$ existuje $\delta > 0$ takové, že pro všechna $y \in \mathbb{B}(x, \delta)$ platí

$$\partial G(y) \subseteq \partial G(x) + \mathbb{B}(0, \varepsilon).$$

Proto je ∂G uzavřená v x , to znamená, že jestliže $\{x^k\} \rightarrow x$, $H^k \in \partial G(x^k)$ a $H^k \rightarrow H$, pak $H \in \partial G(x)$.

Důkaz: Viz [1], str. 71.

Pro reálné funkce může být Clarkův zobecněný gradient charakterizován jistým druhem směrové derivace, kterou definujeme níže.

Definice 3.3. *Nechť je dána funkce $g : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$, kde g je lokálně Lipschitzovská na otevřené množině Ω . Definujeme Clarkovu zobecněnou směrovou derivaci g v x ve směru d , označovanou jako $g^\circ(x; d)$, předpisem*

$$g^\circ(x; d) \equiv \limsup_{\substack{y \rightarrow x \\ t \downarrow 0}} \frac{g(y + td) - g(y)}{t}.$$

Poznámka 3.4. *Jelikož je v definici zobecněné směrové derivace zahrnuta $\lim \sup$, je zřejmé, že $g^\circ(x; d)$ je dobře definována pro každou funkci. Když je g lokálně Lipschitzovská v x jako v našem případě, pak je snadno vidět, že zobecněná směrová derivace $g^\circ(x; d)$ je konečná pro každé d . Ukazuje se, že $g^\circ(x; d)$ a $\partial g(x)$ jsou úzce spojeny:*

$$\partial g(x) = \{ \xi \in \mathbb{R}^n : \xi^T d \leq g^\circ(x; d), \forall d \in \mathbb{R}^n \}.$$

Další vlastnosti Clarkovy zobecněné směrové derivace jsou shrnuty v následující větě.

Věta 3.2. *Nechť je funkce $g : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$ lokálně Lipschitzovská na otevřené množině Ω . Pak platí následující tři tvrzení.*

(a) *Pro každé $x \in \Omega$ je $g^\circ(x; \cdot)$ Lipschitzovská a pozitivně homogenní.*

(b) *Pro každé $(x, d) \in \Omega \times \mathbb{R}^n$,*

$$g^\circ(x; d) = \max \{ \xi^T d : \xi \in \partial g(x) \}.$$

(c) *Jako funkce (x, d) je $g^\circ : \Omega \times \mathbb{R}^n \rightarrow \mathbb{R}$ shora polospojité.*

Důkaz: Viz [1], str. 26 a 28.

Podle definice máme

$$g'(x; d) \leq g^\circ(x; d), \tag{3.2}$$

jestliže směrová derivace vlevo existuje. Jak brzy uvidíme, v mnoha situacích je důležité vyšetřit, zda v (3.2) nastává rovnost.

Definice 3.4. *Nechť je dána funkce $g : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$, kde g je lokálně lipschitzovská na otevřené množině Ω . Řekneme, že g je C -regulární (C podle Clarka) v $x \in \Omega$, jestliže platí podmínky:*

(a) *g má směrovou derivaci v x ve všech směrech,*

(b) *$g^\circ(x; d) = g'(x; d)$ pro všechny směry $d \in \mathbb{R}^n$.*

Kdykoli nehrozí nedorozumění, můžeme jednoduše říci, že funkce g je regulární, jestliže je C -regulární.

Lze ukázat, že konvexní funkce a funkce spojitě diferencovatelné v x jsou C -regulární a jsou tedy nezápornou kombinací regulárních funkcí v témže x . Mimoto, jestliže je g C -regulární funkce, pak máme

$$g'(x; d) = \max \{ \xi^T d : \xi \in \partial g(x) \},$$

což zobecňuje známé tvrzení z konvexní analýzy. Z toho důvodu je třída C -regulárních funkcí poměrně široká. Nicméně existují jednoduché funkce, které nejsou C -regulární. Zde je příklad.

Příklad 3.3. *Uvažujme funkci jedné proměnné $g(x) \equiv -|x|$ a její zobecněnou směrovou derivaci v 0 ve směru $d = 1$. Funkce g má zřejmě derivaci ve všech směrech a $g'(0; 1) = -1$. Na druhou stranu můžeme také snadno nahlédnout, že $g^\circ(0; 1) = 1$, tedy že tato funkce není C -regulární.*

Předchozí příklad, který je formulován v pojmech zobecněné směrové derivace, je další ukázkou skutečnosti, že zobecněný jakobián může být příliš široký pojem. Víme, že obyčejná směrová derivace lokálně lipschitzovské funkce g , pokud existuje, dává přesnou aproximaci g prvního řádu, např.

$$g(x + d) = g(x) + g'(x; d) + o(\|d\|).$$

Příklad 3.3 ukazuje, že zobecněná směrová derivace nemusí mít tuto aproximační vlastnost, která je však zásadní pro skupinu newtonovských metod k řešení rovnice (2.3). Proto pokud bychom chtěli použít zobecněné jakobiány k definování

algoritmu pro řešení systému nehladkých rovnic, museli bychom zúžit třídu funkcí, kterými bychom se zabývali. Definice regularity je prvním pokusem o definování takové třídy funkcí, pro které může být zobecněný jakobián s úspěchem využit k výpočtu.

Pro výpočet zobecněného gradientu funkce je k dispozici množství výpočetných pravidel, které jsou podobné odpovídajícím pravidlům pro diferencovatelné funkce.

Věta 3.3. *Nechť $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, \dots, m$, jsou lokálně lipschitzovské funkce v x .*

(a) *Pro libovolné koeficienty a_i platí*

$$\partial \left(\sum_{i=1}^m a_i g_i \right) (x) \subseteq \sum_{i=1}^m a_i \partial g_i(x),$$

kde rovnost nastává, pokud alespoň jedno g_i je spojitě diferencovatelné v x nebo pokud jsou všechny funkce C -regulární v x a každé a_i je nezáporné. Speciálně $\partial a g(x) = a \partial g(x)$ pro všechny lokálně lipschitzovské funkce g a všechny konstanty a .

(b) *Platí*

$$\partial(g_1 g_2)(x) \subseteq g_2(x) \partial g_1(x) + g_1(x) \partial g_2(x),$$

kde rovnost nastává, pokud g_1, g_2 jsou C -regulární v x , $g_1(x) \geq 0$, a $g_2(x) \geq 0$.

(c) *Jestliže $g_2(x) \neq 0$, pak*

$$\partial \left(\frac{g_1}{g_2} \right) \subseteq \frac{g_2(x) \partial g_1(x) - g_1(x) \partial g_2(x)}{g_2^2(x)},$$

kde rovnost nastane, pokud $g_1, -g_2$ jsou C -regulární v x , $g_1(x) \geq 0$ a $g_2(x) > 0$.

(d) Pro bodově maximovou funkci

$$g(x) \equiv \max\{g_i(x), i = 1, \dots, m\}$$

platí, že

$$\partial g(x) \subseteq \text{conv}\{\partial g_i(x) : i \in I(x)\},$$

kde $I(x) \equiv \{i : g_i(x) = g(x)\}$. Navíc rovnost v inkluzi platí, pokud všechny g_i jsou C -regulární v x .

Důkaz: Viz [1], str. 39, 40, 47 a 48.

Bod (d) je zvláště významný, protože poskytuje důležitý zdroj nediferencovatelných funkcí (maximum konečného počtu funkcí není diferencovatelné všude, i když všechny funkce g_i diferencovatelné jsou) a nemá žádnou analogii v hladkém případě. Pokud nejsou splněny přísnější podmínky, početní pravidla nám dávají pouze inkluzi. Tento fakt představuje hranici v možnostech snadného výpočtu zobecněného gradientu. Následuje jednoduchý příklad rozdílu, který může existovat mezi množinami na levé a pravé straně v inkluzích z věty 3.3.

Příklad 3.4. Necht $g_1(x) = |x|$ a $g_2(x) = -|x|$. Tyto dvě funkce jsou všude lokálně lipschitzovské a jejich součet

$$g(x) \equiv g_1(x) + g_2(x)$$

je identicky roven nule. Zobecněný gradient g je proto všude 0. Odhadněme zobecněný gradient g v nule za pomoci bodu (a) z věty 3.3. Budeme-li mít na paměti pravidla pro sčítání množin, dostaneme množinu $[-1, 1] + [-1, 1] = [-2, 2]$, což je zřejmě velmi odlišné od $\partial g(0)$. Ovšem inkluze uvedená ve větě 3.3 platí: $\partial g(0) \subseteq \partial g_1(0) + \partial g_2(0)$. Všimněte si, že g_2 není C -regulární v nule, jak ukazuje příklad 3.3.

Výsledky seřazené v předchozí větě mohou být chápány jako speciální případy následující věty o složené funkci.

Věta 3.4. *Nechť $f = g \circ G$, kde $G : \mathbb{R}^n \rightarrow \mathbb{R}^m$ je lokálně lipschitzovská v x a $g : \mathbb{R}^m \rightarrow \mathbb{R}$ je lokálně lipschitzovská v $G(x)$. Pak f je lokálně lipschitzovská v x a*

$$\partial f(x) \subseteq \text{conv} \{ \xi = H^T \zeta : H \in \partial G(x), \zeta \in \partial g(G(x)) \}.$$

Navíc platí rovnost a konvexní obal je nadbytečný, pokud je splněna alespoň jedna z následujících dvou podmínek:

(a) *g je spojitě diferencovatelná v $G(x)$;*

(b) *g je C -regulární v $G(x)$ a G je spojitě diferencovatelná v x (v tomto případě je f C -regulární v x).*

Důkaz: Viz [1], str. 43.

Analogicky k hladkému případu můžeme uvažovat částečně zobecněné gradienty.

Poznámka 3.5. *Předpokládáme, že je dána funkce $g : \mathbb{R}^p \times \mathbb{R}^q \rightarrow \mathbb{R}$. Značením $\partial_x g(x, y)$ budeme rozumět zobecněný gradient $g(\cdot, y)$ v x , a značením $\partial_y g(x, y)$ pak zobecněný gradient $g(x, \cdot)$ v y . Obecně je buď $\partial g(x, y)$ obsažen v $\partial_x g(x, y) \times \partial_y g(x, y)$ nebo obráceně. Nicméně vždy platí, že*

$$\partial_x g(x, y) \subseteq \Pi_x \partial g(x, y) \quad \text{a} \quad \partial_y g(x, y) \subseteq \Pi_y \partial g(x, y),$$

kde Π_x (Π_y) označují kanonickou projekci na podprostor proměnných x (y). Dále je-li g regulární v (x, y) , pak platí, že

$$\partial g(x, y) \subseteq \partial_x g(x, y) \times \partial_y g(x, y).$$

Zobecněné gradienty nám dovolují rozšířit známý klasický výsledek pro hladké funkce.

Věta 3.5. *Předpokládejme, že $g : \mathbb{R}^n \rightarrow \mathbb{R}$ je lipschitzovská v okolí svého nepodmíněného lokálního minima x . Pak*

$$0 \in \partial g(x). \tag{3.3}$$

Důkaz: Viz [1], str. 38.

Poznámka 3.6. Vektor x splňující (3.3) nazýváme C -stacionární bod g , kde C znamená Clarkův. Tudiž každé lokální minimum lokálně lipschitzovské funkce je C -stacionárním bodem.

Můžeme také vyslovit věty o střední hodnotě, nejdříve pro skalární funkci.

Věta 3.6. Nechť je dána funkce $g : \mathbb{R}^n \rightarrow \mathbb{R}$, která je lipschitzovská na otevřené množině obsahující úsečku $[x, y]$. Pak existuje bod $z \in (x, y)$ takový, že

$$g(y) = g(x) + \xi^T(y - x),$$

pro nějaké ξ náležející do $\partial g(z)$.

Důkaz: Viz [1], str. 41.

Ve výše uvedených větách jsme se omezili na reálné skalární funkce, ale mnohé z nich je možno vyslovit také pro funkce vektorové. Nejdříve uvedeme větu, která dává do souvislosti zobecněný jakobián vektorové funkce a zobecněný gradient jejích složek.

Věta 3.7. Nechť $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$ je lokálně lipschitzovská funkce na otevřené množině Ω . Jestliže $x \in \Omega$, pak

$$\partial G(x) \subseteq (\partial_1 G(x) \times \partial_2 G(x) \times \dots \times \partial_m G(x))^T.$$

Důkaz: Viz [1], str. 71.

Výše uvedená inkluze je obvykle rovností, pokud nediferencovatelnost různých složek G je „nesouvisející“, zatímco v opačném případě je inkluze obvykle ostrá. Předvedeme to dvěma jednoduchými příklady.

Příklad 3.5. Uvažujme funkci

$$F(x_1, x_2) \equiv \begin{pmatrix} |x_1| + 2x_2 \\ -x_1 + |x_2| \end{pmatrix}.$$

Máme

$$\partial F(0) = \left\{ \begin{pmatrix} \alpha & 2 \\ -1 & \beta \end{pmatrix}, \forall \alpha \in [-1, 1], \forall \beta \in [-1, 1] \right\}.$$

V tomto případě je inkluze z věty 3.7 ve skutečnosti rovností.

Uvažujme jinou funkci

$$F(x_1, x_2) \equiv \begin{pmatrix} |x_1| - \tan x_2 \\ -|x_1| + e^{x_2} \end{pmatrix}.$$

Máme

$$\partial F(0) = \left\{ \begin{pmatrix} \alpha & -1 \\ -\alpha & 1 \end{pmatrix}, \forall \alpha \in [-1, 1] \right\},$$

zatímco

$$\partial F_1(0) \times \partial F_2(0) = \left\{ \begin{pmatrix} \alpha & -1 \\ \beta & 1 \end{pmatrix}, \forall \alpha \in [-1, 1], \forall \beta \in [-1, 1] \right\}.$$

V tomto případě inkluze z věty 3.7 platí jako ostrá.

Následující věta rozšiřuje větu 3.6 pro vektorové funkce.

Věta 3.8. *Nechť je funkce $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$ lipschitzovská na otevřené množině Ω obsahující úsečku $[x, y]$. Pak existuje m bodů z^i z intervalu (x, y) a m koeficientů $\alpha_i \geq 0$, jejichž součet je jedna, takových, že*

$$G(y) = G(x) + \sum_{i=1}^m \alpha_i H_i(y - x),$$

kde H_i náleží do $\partial G(z^i)$ pro každé i .

Důkaz: Viz [1], str. 72.

Klasický tvar věty o střední hodnotě pro vektorové funkce třídy C^1 je obvykle uveden v integrální formě. Alternativní tvar uvedený výše, který zahrnuje jednoduchou sumaci, není příliš známý dokonce ani v hladkém případě. Integrální verze

existuje také pro případ nehladký, nicméně pro náš záměr je tvar prezentovaný výše vhodnější.

Následující výsledek ukazuje důležitou spojitost mezi směrovou derivací a Clarkovým zobecněným jakobiánem.

Věta 3.9. *Nechť je funkce $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$, kde Ω je otevřená, B -diferencovatelná v bodě $x \in \Omega$. Pro každý vektor $d \in \mathbb{R}^n$ existuje $H \in \partial G(x)$ takové, že $G'(x; d) = Hd$.*

Důkaz: Viz [2], str. 636.

Uvedením věty o implicitní funkci pro funkci lokálně lipschitzovskou můžeme nyní uzavřít tuto kapitolu. Vrátime se k našemu záměru popsat metody řešení nehladkých rovnic, k němuž využijeme výše představené nástroje.

4 Princip nehladké Newtonovy metody

Nyní, když máme k dispozici aparát Clarkova zobecněného jakobiánu, zaměřme se na princip Newtonovy metody a pokusme se vybudovat její nehladkou obdobu.

Nechť je dána funkce $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$. Vývoj metod Newtonova typu pro řešení rovnice (2.2),

$$G(x) = 0,$$

kde G je nehladká, je motivován klasickým Newtonovým algoritmem pro spojitě diferencovatelné funkce. Tento algoritmus je prototypem mnoha lokálních, rychlých algoritmů pro řešení hladkých rovnic. Takové algoritmy mají výbornou rychlost konvergence v okolí nulového bodu G , ale mohou selhat, pokud je počáteční bod příliš daleko od řešení. Základní myšlenkou v obecné metodě Newtonova typu je nahradit funkci G aproximací závisující na aktuální iteraci, což má za výsledek, že aproximovanou úlohu lze řešit mnohem snadněji. Řešení této aproximace se pak považuje za nový počáteční bod a celý postup se opakuje.

Jestliže je funkce G spojitě diferencovatelná, nabízí se přirozená aproximace, a to linearizace v aktuálním bodě. Přesněji, je-li dán bod x^k , můžeme zformulovat lineární aproximaci

$$G(x^k) + JG(x^k)(x - x^k) \tag{4.1}$$

a vypočítat x^{k+1} jako nulový bod této lineární aproximace. Pokusme se najít vysvětlení rychlé konvergence klasické Newtonovy metody. Jako první krok předpokládejme, že počáteční bod x^0 je dostatečně blízko k nulovému bodu x^* funkce G , a že $JG(x^*)$ je regulární. Obecně, je-li dán bod x^k dostatečně blízko k x^* , máme pomocí Taylorova rozvoje

$$0 = G(x^*) = G(x^k) + JG(x^k)(x^* - x^k) + o_{x^k}(\|x^k - x^*\|),$$

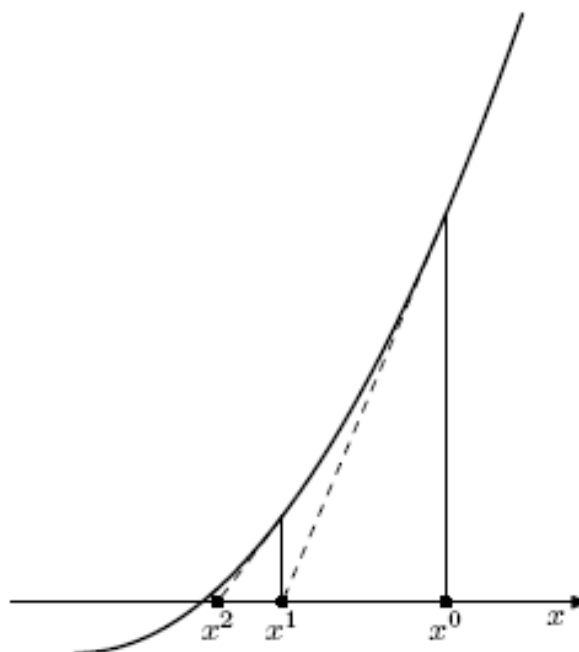
kde píšeme o_{x^k} , abychom zdůraznili, že „malé o “ v Taylorově rozvoji závisí především na bodě x^k . Pokud vynásobíme výše uvedený výraz $JG(x^k)^{-1}$ (díky spojitosti lze v okolí x^* jakobián JG invertovat), dostaneme

$$x^k - x^* - JG(x^k)^{-1}G(x^k) = o_{x^k}(\|x^k - x^*\|).$$

Pokud lze ověřit, že funkce „malé o “ je nezávislá na x^k , poslední rovnice okamžitě dává

$$x^{k+1} - x^* = (x^k - JG(x^k)^{-1}G(x^k)) - x^* = o(\|x^k - x^*\|).$$

Tento výraz je základním kamenem toho, že Newtonova posloupnost $\{x^k\}$ je dobře definovaná, konverguje k x^* a tato konvergence je Q-superlineární (viz definici 4.1). Ukazuje se, že „malé o “ v Taylorově rozvoji je vskutku nezávislé na x^k (viz větu 4.1). Obrázek 3 ilustruje tuto iteraci.



Obrázek 3: Ilustrace hladké Newtonovy metody

Definice 4.1. *Nechť $\{x^k\} \subset \mathbb{R}^n$ je posloupnost vektorů jdoucí k limitě $x \neq x^k$ pro všechna k . Řekneme, že řád konvergence je (nejméně)*

(a) *Q-lineární, jestliže*

$$\limsup_{k \rightarrow \infty} \frac{\|x^{k+1} - x\|}{\|x^k - x\|} < \infty;$$

(b) *Q-superlineární, jestliže*

$$\limsup_{k \rightarrow \infty} \frac{\|x^{k+1} - x\|}{\|x^k - x\|} = 0;$$

(c) *Q-kvadratický, jestliže*

$$\limsup_{k \rightarrow \infty} \frac{\|x^{k+1} - x\|}{\|x^k - x\|^2} < \infty;$$

(d) *R-lineární, jestliže*

$$0 < \limsup_{k \rightarrow \infty} (\|x^k - x\|)^{1/k} < 1.$$

V každém případě říkáme, že $\{x^k\}$ konverguje k x nejméně Q-lineárně, respektive Q-superlineárně, Q-kvadraticky, R-lineárně.

Vrátíme se nyní k Newtonově postupu. Fungování tohoto procesu umožňují dvě vlastnosti: lineární model (4.1) poskytuje „dobrou“ aproximaci G v blízkosti x^k a výsledná lineární rovnice je řešitelná, protože $JG(x^k)$ je invertibilní. Tyto dvě vlastnosti závisí na spojitě diferencovatelnosti G a na regularitě $JG(x^*)$. Naším cílem je vytvořit analogickou metodu a zachovat přitom tyto klíčové vlastnosti, aniž bychom požadovali spojitou diferencovatelnost G .

V této kapitole předložíme několik obecných podmínek, které zachycují základ Newtonovy metody a které lze aplikovat na nediferencovatelné funkce. Za tímto účelem položíme $d = x - x^k$ a přepíšme (4.1) jako

$$G(x^k) + JG(x^k)d. \tag{4.2}$$

Pak můžeme získat x^{k+1} jako $x^k + d^k$, kde d^k hledáme jako nulový bod (4.2). Proto d^k reprezentuje posunutí, které uděláme z x^k do x^{k+1} . Jedná se pouze o formální změnu, kterou provádíme kvůli snažšímu značení. Všimněme si také, že $JG(x^k)d$ je „dobrá“ aproximace $G(x^k + d) - G(x^k)$, alespoň pro malé d . Když je G nediferencovatelné, nemusí Jakobián G v bodě x^k dokonce ani existovat, a i když existuje, není žádný důvod předpokládat, že ho můžeme použít k získání dobré aproximace hodnoty G v nejbližších bodech. Chceme-li zobecnit (4.2), předpokládejme, že Newtonův model je dán pomocí

$$G(x^k) + A(x^k, d), \quad (4.3)$$

kde $A(x^k, d)$ je model $G(x^k + d) - G(x^k)$ okolo $d = 0$, takže (4.3) lze považovat za aproximaci $G(x^k + d)$. Abychom dosáhli dostatečné obecnosti, předpokládejme, že existuje skupina aproximací $\mathcal{A}(x)$, kde každý prvek $A(x, \cdot) \in \mathcal{A}(x)$ je funkce z \mathbb{R}^n do \mathbb{R}^n , která může být použita v (4.3). Příkladem je lineární model $G(x^k) + Hd$, kde H je libovoný prvek zobecněného jakobiánu $\partial G(x)$. V tomto případě $\mathcal{A} = \partial G$. Dále v této kapitole se budeme zabývat případy, ve kterých je vhodné předpokládat, že v každém kroku máme více než jeden model. Pokud chceme používat tyto modely k vytvoření lokální Newtonovy metody pro nehladké rovnice, musíme zřejmě zavést nějaké adekvátní podmínky. Následující definice obsahuje základní vlastnosti, které budeme potřebovat.

Definice 4.2. *Nechť G je lokálně lipschitzovská funkce z otevřené množiny $\Omega \subset \mathbb{R}^n$ do \mathbb{R}^m . Říkáme, že G má Newtonovu aproximaci v bodě $\bar{x} \in \Omega$, jestliže existuje okolí $\Omega' \subseteq \Omega$ a funkce $\Delta : (0, \infty) \rightarrow [0, \infty)$, taková, že*

$$\lim_{t \downarrow 0} \Delta(t) = 0, \quad (4.4)$$

a navíc taková, že pro každý bod x z Ω' existuje třída $\mathcal{A}(x)$ funkcí, z nichž každá zobrazuje \mathbb{R}^n do \mathbb{R}^m a splňuje následující dvě vlastnosti:

(a) $A(x, 0) = 0$ pro každé $A(x, \cdot) \in \mathcal{A}(x)$,

(b) pro každé $x \in \Omega'$ různé od \bar{x} a pro každé $A(x, \cdot) \in \mathcal{A}(x)$ platí

$$\frac{\|G(x) + A(x, \bar{x} - x) - G(\bar{x})\|}{\|x - \bar{x}\|} \leq \Delta(\|x - \bar{x}\|).$$

\mathcal{A} nazýváme (Newtonovo) aproximační schéma pro G v \bar{x} . Pokud je požadavek (b) zesílen na

(b') existuje kladná konstanta L' taková, že pro každé $x \in \Omega'$ různé od \bar{x} a pro všechny $A(x, \cdot) \in \mathcal{A}(x)$ platí

$$\frac{\|G(x) + A(x, \bar{x} - x) - G(\bar{x})\|}{\|x - \bar{x}\|^2} \leq \Delta(\|x - \bar{x}\|),$$

pak říkáme, že F má silnou Newtonovu aproximaci v \bar{x} a že \mathcal{A} je silné Newtonovo aproximační schéma. Navíc, pokud je splněna následující dodatečná podmínka:

(c) $m = n$, a \mathcal{A} je třída stejnoměrně lipschitzovských homeomorfizmů na Ω' , čímž máme na mysli, že existují kladné konstanty $L_{\mathcal{A}}$ a $\varepsilon_{\mathcal{A}}$ takové, že pro každé $x \in \Omega'$ a pro každé $A(x, \cdot) \in \mathcal{A}$ najdeme dvě otevřené množiny U_x, V_x , obě obsahující $\mathbb{B}(0, \varepsilon_{\mathcal{A}})$, takové, že $A(x, \cdot)$ je lipschitzovský homeomorfismus zobrazující U_x na V_x , kde $L_{\mathcal{A}}$ je lipschitzovský modul inverzního zobrazení restrikce $A(x, \cdot)|_{U_x}$,

říkáme, že silná Newtonova aproximace i silné aproximační schéma \mathcal{A} jsou regulární. Jestliže \mathcal{A} obsahuje pouze jeden prvek pro každé $x \in \Omega'$, říkáme, že G připouští jednoznačnou silnou Newtonovu aproximaci v \bar{x} a že aproximační schéma \mathcal{A} je jednoznačné.

Poznamenejme, že požadavek (c) může být splněn pouze pokud $m = n$, ale definice Newtonova aproximačního schématu je formulována také pro případ $m \neq n$. Pro naše účely však plně postačí $m = n$. Podmínka (a) nám říká, že model (4.2) souhlasí s $G(x)$ pro $d = 0$, a je zcela přirozená. Podmínka (b) (a (b')) stanovuje, že model (4.2) dobře aproximuje hodnotu $G(\bar{x})$. Všimněte si, že tento požadavek

je velmi slabý, a nepožaduje, aby model (4.2) byl dobrou aproximací $G(x)$ na celém okolí daného bodu. Tato podmínka bude mít stejnou roli jako Taylorova věta pro hladké funkce. Konečně, podmínka (c) je vlastně podmínka regularity a je v zásadě nutná pro garanci řešitelnosti Newtonovy rovnice v okolí \bar{x} a pro zaručení Q-superlineární třídy konvergence. Je blízká podmínce regularity jakobiánu funkce G , když G je spojitě diferencovatelná.

Poznámka 4.1. *Podmínky (b) a (b') mohou být ekvivalentně vyjádřeny jako*

$$\lim_{\substack{x \rightarrow \bar{x} \\ A(x, \cdot) \in \mathcal{A}(x)}} \frac{\|G(x) + A(x, \bar{x} - x) - G(\bar{x})\|}{\|x - \bar{x}\|} = 0,$$

respektive

$$\limsup_{\substack{x \rightarrow \bar{x} \\ A(x, \cdot) \in \mathcal{A}(x)}} \frac{\|G(x) + A(x, \bar{x} - x) - G(\bar{x})\|}{\|x - \bar{x}\|} < \infty.$$

Tento způsob formulace podmínek (b) a (b') je pravděpodobně náročnější, nicméně definice 4.2 se pak lépe přizpůsobuje některým z následujících výsledků.

Pro zjednodušení označíme inverzi funkce $A(x, \cdot)|_{\mathcal{U}_x}$ jako $A^{-1}(x, \cdot)$ a vypustíme definiční obor \mathcal{V}_x a obor hodnot \mathcal{U}_x inverzní funkce. Proto pro všechny $y \in \mathcal{V}_x$ píšeme

$$A^{-1}(x, y) = (A(x, \cdot)|_{\mathcal{U}_x})^{-1}(y).$$

Tato notace je přirozená a neměla by být nijak matoucí.

S pomocí výše uvedené definice představíme následující přirozené rozšíření hladké Newtonovy metody pro řešení nehladkých rovnic a následně ukážeme, že rozšířená metoda si uchovává hlavní znak, který charakterizuje původní metodu, tj. rychlou lokální konvergenci.

Algoritmus 4.1 (Nehladká Newtonova metoda).

Data: $x^0 \in \mathbb{R}^n$ a $\varepsilon > 0$.

Krok 1: Položíme $k = 0$.

Krok 2: Jestliže $G(x^k) = 0$, stop.

Krok 3: Vybereme $A(x^k, \cdot)$ z $\mathcal{A}(x)$ a najdeme vektor d^k z $\mathbb{B}(0, \varepsilon)$ takové, že

$$G(x^k) + A(x^k, d^k) = 0. \quad (4.5)$$

Krok 4: Položíme $x^{k+1} \equiv x^k + d^k$ a $k \leftarrow k + 1$. Jdeme na krok 2.

Je třeba říci pár slov o ukončovacím kritériu v kroku 2. Algoritmus 4.1 skončí v konečném počtu kroků jen tehdy, když dosáhne přesně nulového bodu G . Prakticky tento druh konečného zastavení nikdy nenastane. Užitečné zastavovací kritérium je proto následující:

$$\|G(x^k)\| \leq \text{předepsaná tolerance.}$$

Pokud platí, bod x^k je při ukončení akceptován jako dostačující aproximace nulového bodu G .

Tato poznámka o ukončovacím pravidlu se vztahuje na všechny představené algoritmy. V praktické implementaci jsou všechny iterační algoritmy ukončeny podle takového zastavovacího kritéria. Ovšem při sestavování těchto algoritmů budeme pokračovat v používání kritérií pro přesné řešení. Hlavním předmětem konvergenční analýzy je demonstrovat, že takové kritérium je splněno asymptoticky a v rozumně rychlé třídě konvergence nekonečnou posloupností bodů generovanou algoritmem.

Jediný formální rozdíl mezi nehladkou a klasickou Newtonovou metodou je, že v Newtonově rovnici (4.5) v kroku 3 se nepožaduje jediné řešení. Takto musí být specifikovány jisté dodatečné požadavky na řešení, například že náleží do vhodně malého okolí počátku. Dokonce i v tomto omezeném okolí může mít rovnice (4.5) více než jedno řešení. Nicméně v mnoha významných případech jsou lokální modely $A(x^k, \cdot)$ vlastně homeomorfismy a proto má každá Newtonova rovnice právě

jedno řešení. V situaci, kdy $A(x^k, \cdot)$ není homeomorfismus a jednoznačnost řešení (4.5) je ohrožena, je restrikce na malé okolí $\mathbb{B}(0, \varepsilon)$ zásadní pro konvergenci celého algoritmu.

Dosud nebyla explicitně zodpovězena otázka obtížnosti řešení Newtonovy rovnice (4.5), ale rozumí se, že tato rovnice by měla být snáze řešitelná než původní rovnice (2.3), jinak by celý Newtonův proces nebyl výhodný.

Následující věta zavádí klíčové teoretické vlastnosti algoritmu 4.1. Je třeba zdůraznit, že věta se týká lokální konvergence, což znamená předpoklad, že počáteční bod x^0 je vybrán z dostatečně malého okolí hledaného, ale neznámého řešení. Tento předpoklad je také potřeba v rozšířené větě 4.2, která se týká „neexaktní“ verze metody. Připomeňme, že ε_A je kladná konstanta zmíněná v definici 4.2.

Věta 4.1. *Nechť $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$, kde Ω je otevřená, je lokálně lipschitzovská funkce v okolí $x^* \in \Omega$, splňující $G(x^*) = 0$. Předpokládejme, že G připouští regulární Newtonovu aproximaci A v x^* . Pro každé $\varepsilon \in (0, \varepsilon_A]$ existuje okolí $\mathbb{B}(x^*, \delta)$ bodu x^* takové, že jestliže x^0 náleží do $\mathbb{B}(x^*, \delta)$, pak Newtonův algoritmus 4.1 generuje jedinou posloupnost $\{x^k\}$, která konverguje Q -superlineárně k x^* . Pokud Newtonova aproximace A je silná, třída konvergence je Q -kvadratická.*

Důkaz: Viz [2], str. 645.

V praktickém použití může být výpočetně velmi náročné řešit Newtonovu rovnici (4.5) exaktně. To je pravda také když model $A(x, \cdot)$ je lineární (jako v hladkém případě), ale hodně velký. Pak je užitečné zvážit řešení rovnice (4.5) neexaktně. Ukazuje se, že všechny vlastnosti, které jsme určili u exaktního nehladkého Newtonova algoritmu 4.1, zůstávají v platnosti, jestliže řešíme Newtonovu rovnici (4.5) přiměřeně přesným způsobem, který bude stanoven podmínkami (4.6) a (4.7). Tyto dvě podmínky poskytují jednotnou strukturu pro všechny neexaktní nehladké Newtonova schémata.

Algoritmus 4.2 (Neexaktní nehladká Newtonova metoda).

Data: $x^0 \in \mathbb{R}^n$, $\varepsilon > 0$ a posloupnost nezáporných skalárů $\{\eta_k\}$.

Krok 1: Položíme $k = 0$.

Krok 2: Jestliže $G(x^k) = 0$, stop.

Krok 3: Vybereme $A(x^k, \cdot) \in \mathcal{A}(x)$ a najdeme směr $d^k \in \mathbb{B}(0, \varepsilon)$ takový, že

$$G(x^k) + A(x^k, d^k) = r^k, \quad (4.6)$$

kde r^k je vektor vyhovující

$$\|r^k\| \leq \eta_k \|G(x^k)\|. \quad (4.7)$$

Krok 4: Položíme $x^{k+1} \equiv x^k + d^k$ a $k \leftarrow k + 1$. Jdeme na krok 2.

Je záhodno dále diskutovat neexaktní pravidlo (4.7). Zhruba řečeno toto pravidlo stanovuje, že neexaktnost vektrem d^k jako aproximativního řešení Newtonovy rovnice je úměrná k zbytkové normě $\|G(x^k)\|$ v bodě x^k . Toto pravidlo je zásadní pro rychlou konvergenci posloupnosti $\{x^k\}$ vytvořenou výše uvedenou neexaktní Newtonovou metodou, viz větu 4.2. Než uvedeme tuto větu, vyslovíme lemma, které zaručuje existenci směru d^k v kroku 3 algoritmu 4.2 pro každý vektor r^k vyhovující (4.7).

Lemma 4.1. *Nechť $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$, kde Ω je otevřená, je lokálně lipschitzovská funkce v okolí $x^* \in \Omega$ splňující $G(x^*) = 0$. Předpokládejme, že G připouští regulární Newtonovu aproximaci A v x^* . Pak pro každé $\varepsilon \in (0, \varepsilon_A]$ a každé $\bar{\eta} > 0$ existuje okolí $\mathbb{B}(x^*, \delta)$ bodu x^* takové, že pro každý vektor $x^k \in \mathbb{B}(x^*, \delta)$, každý skalár $\eta_k \in (0, \bar{\eta}]$ a každý vektor r^k vyhovující (4.7) má rovnice (4.6) jediné řešení $d^k \in \mathbb{B}(0, \varepsilon)$.*

Důkaz: Viz [2], str. 647.

Výše uvedené lemma je koncepčně užitečné, ovšem jeho předpoklady jsou opačné, než jak je zamýšlen algoritmus 4.2. Při implementaci tohoto neexaktního algoritmu není dán residuální vektor r^k a priori. Místo toho v kroku 3 počítáme

vhodný směr d^k tak, aby asociovaný residuální vektor $r^k \equiv G(x^k) + A(x^k, d^k)$ splňoval (4.7). Řešením může být například použití iterační metody na rovnici $0 = G(x^k) + A(x^k, d)$ a ukončení této metody po konečně mnoha vnitřních iteracích, kdy je jako zastavovací kritérium použito (4.7). Obecně existuje mnoho vektorů d^k , které mohou být vybrány v kroku 3 algoritmu 4.2. To je v kontrastu s algoritmem 4.1, kde je podle věty 4.1 vektor d^k jediný. Další věta se zabývá libovolnou posloupností $\{x^k\}$ generovanou algoritmem 4.2. Je zaměřena ani ne tak na existenci nebo jednoznačnost takové posloupnosti, ale spíše na její konvergenci.

Věta 4.2. *Nechť $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$, kde Ω je otevřená, je lokálně lipschitzovská funkce v okolí $x^* \in \Omega$, splňující $G(x^*) = 0$. Předpokládejme, že G připouští regulární Newtonovu aproximaci A v x^* . Pak existuje kladné číslo $\bar{\eta}$ takové, že když $\eta_k \leq \bar{\eta}$ pro každé k , pak pro každé $\varepsilon \in (0, \varepsilon_A]$ existuje okolí $\mathbb{B}(x^*, \delta)$ bodu x^* takové, že pokud x^0 náleží do $\mathbb{B}(x^*, \delta)$, neexaktní Newtonova metoda 4.2 je dobře definovaná a každá posloupnost $\{x^k\}$ generovaná touto metodou konverguje Q -lineárně k x^* . Navíc, jestliže posloupnost $\{\eta_k\} \rightarrow 0$, třída konvergence $\{x^k\}$ k x^* je Q -superlineární. Konečně, jestliže Newtonova aproximace A je silná a pro některé $\tilde{\eta} > 0$ platí $\eta_k \leq \tilde{\eta} \|G(x^k)\|$ pro všechna k , pak třída konvergence je Q -kvadratická.*

Důkaz: Viz [2], str. 648.

Dále uvážíme klasický případ, kdy G je spojitě diferencovatelná blízko nulového bodu x^* . Nechť $A(x^k, d)$ je model daný (4.3). Algoritmus 4.1 se pak zredukuje na hladkou Newtonovu metodu. Navíc věta 4.1 snadno dává dobře známý konvergenční výsledek pro tuto Newtonovu metodu. Abychom to ukázali, ověříme vlastnosti (a), (b), (b') a (c) za předpokladu spojitě diferencovatelnosti G . První vlastnost (a) je splněna triviálně. Předpokládejme nyní, že $JG(x^*)$ je regulární. Pak existuje okolí Ω bodu x^* a kladná konstanta L takové, že

$$\|JG(x)\| \leq L \quad \text{a} \quad \|JG(x)^{-1}\| \leq L,$$

pro každé $x \in \Omega$. Pak je jasné, že pro všechny x^k z Ω jsou $A(x^k, d)$ i jeho inverze lipschitzovské homeomorfismy s lipschitzovskou konstantou L , a tudíž je podmínka (c) z definice 4.2 jasně splněna. Zbývá tedy ověřit (b) a (b') z definice 4.2, což se redukuje na elementární aplikaci věty o střední hodnotě pro vektorové funkce. Formálně bude tento výsledek uveden v obecnějším tvaru v následující větě, na kterou bude možno se dále odvolávat.

Věta 4.3. *Nechť $G : \mathbb{R}^n \rightarrow \mathbb{R}^m$ je spojitě diferencovatelná v okolí Ω bodu \bar{x} . Pak existuje neklesající funkce $\Delta : (0, \infty) \rightarrow [0, \infty)$ s vlastností*

$$\lim_{t \downarrow 0} \Delta(t) = 0$$

taková, že

$$\|G(x) + JG(x)(z - x) - G(z)\| \leq \|x - z\| \Delta(\|x - z\|)$$

pro všechna $z, x \in \Omega$. Navíc, jestliže je JG lipschitzovské v okolí \bar{x} , pak existují okolí $\Omega' \subseteq \Omega$ bodu \bar{x} a kladná konstanta L' tak, že

$$\|G(x) + JG(x)(z - x) - G(z)\| \leq L' \|x - z\|^2$$

pro všechna $z, x \in \Omega'$.

Důkaz: Viz [2], str. 651.

Před tím, než se zaměříme na funkce po částech hladké, pro které mohou být získány rychlé lokálně konvergentní algoritmy z Newtonových schémat popsanych dosud, přidáme několik dalších vět, které nám pomohou pochopit podstatu předpokladů. První věta ukazuje, že pokud G připouští regulární Newtonovu aproximaci v řešení x^* , pak x^* je izolované řešení. Je obdobou klasické věty, že jestliže je jakobián v bodě řešení diferencovatelného systému rovnic regulární, pak je řešení izolované. Připomeňme, že řešení x je izolované, pokud existuje jeho okolí tak, že x je jediným řešením na tomto okolí.

Věta 4.4. *Nechť $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$, kde Ω je otevřená, je lokálně lipschitzovská funkce v nulovém bodě $x^* \in \Omega$ zobrazení G . Jestliže má G Newtonovu aproximaci*

$A(x)$ v x^* , pro kterou existuje konstanta $c > 0$, okolí N bodu x^* a okolí počátku U takové, že pro všechna $x \in N$ existuje prvek $A(x) \in A(x)$ splňující

$$\|A(x, d)\| \geq c\|d\|, \quad \forall d \in U,$$

pak existuje okolí N' bodu x^* a kladná konstanta c' takové, že

$$\|x - x^*\| \leq c'\|G(x)\|, \quad \forall x \in N',$$

příčemž x^* je lokálně jediný nulový bod G .

Důkaz: Viz [2], str. 652.

Poznámka 4.2. Předpoklady věty 4.4 zřejmě platí, pokud je Newtonova aproximace A regulární. Proto existence regulární Newtonovy aproximace implikuje bodově ohraničenou chybu.

Při definování Newtonova aproximačního schématu je obvykle nejtežším bodem ověření jeho regularity. Věta 4.5 ukazuje, že za vhodných předpokladů můžeme vyvodit regularitu Newtonova aproximačního schématu z bodové regularity v báзовých vektorech. Níže uvedené lemma, které je zobecněním klasického konečně-dimenzionálního Banachova perturbačního lemmatu, je úhelným kamenem pro důkaz této věty.

Lemma 4.2. Nechť A a A' jsou dvě funkce zobrazující množinu $U \subset \mathbb{R}^n$ do \mathbb{R}^n takové, že A je lipschitzovský homeomorfismus a $A - A'$ je lipschitzovské na U . Nechť $L > 0$, respektive $L' > 0$, je lipschitzovská konstanta A , respektive $A - A'$. Pokud $d^* \in U$ a $\delta > 0$ jsou takové, že

(a) $uz \mathbb{B}(d^*, \delta) \subseteq U$,

(b) $uz \mathbb{B}(A(d^*), \delta/L) \subseteq A(U)$,

(c) $LL' < 1$,

pak A' je lipschitzovský homeomorfismus z \mathcal{U} na $A'(U)$ a jeho inverze $(A')^{-1}$ má lipschitzovskou konstantu

$$\frac{L}{1 - LL'} > 0.$$

Navíc

$$B\left(A'(d^*), \frac{(1 - LL')\delta}{L}\right) \subseteq A'(U).$$

Důkaz: Viz [2], str. 653.

Následující věta nám dává postačující podmínku pro regularitu Newtonovy aproximace. Jak bylo zmíněno výše, klíčové je zaručit platnost předpokladů lemmatu 4.2.

Věta 4.5. *Nechť $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ je lokálně lipschitzovská funkce na otevřené množině Ω obsahující vektor \bar{x} . Předpokládejme, že \mathcal{A} je Newtonovo aproximační schéma funkce G v \bar{x} , pro které existují tři kladné konstanty $\varepsilon_1, \varepsilon_2$ a L , splňující*

(A) *pro každé $A(\bar{x}, \cdot) \in \mathcal{A}(\bar{x})$ najdeme dvě množiny U, V obsahující $\mathbb{B}(0, \varepsilon_1)$, respektive $\mathbb{B}(0, \varepsilon_1)$, takové, že $A(\bar{x}, \cdot)$ je lipschitzovský homeomorfismus z U do V a $A^{-1}(\bar{x}, \cdot)$ má lipschitzovskou konstantu L .*

Předpokádejme dále, že existuje funkce $L' : (0, \infty) \rightarrow [0, \infty)$, pro kterou platí $\lim_{t \downarrow 0} L'(t) = 0$, a okolí \mathcal{N} bodu \bar{x} takové, že je splněna právě jedna z následujících dvou podmínek:

- (a) *pro každé $x \in \mathcal{N}$ a každé $A(x, \cdot) \in \mathcal{A}(x)$ existuje prvek $A(\bar{x}, \cdot) \in \mathcal{A}(\bar{x})$ takový, že $A(x, \cdot) - A(\bar{x}, \cdot)$ je lipschitzovské s konstantou $L'(\|x - \bar{x}\|)$ na U ;*
- (b) *pro každé $x \in \mathcal{N}$, $\mathcal{A}(x) = \{A(x, \cdot)\}$ je jednoznačné a $A(x, \cdot) - \tilde{A}(\bar{x}, \cdot)$ je lipschitzovské s konstantou $L'(\|x - \bar{x}\|)$ na U , kde*

$$\tilde{A}(x, d) \equiv A(x, \bar{x} - x + d).$$

Pak je aproximační schéma \mathcal{A} regulární.

Důkaz: Viz [2], str. 655.

5 Aplikace na funkce po částech hladké

Jedna z oblastí aplikace nehladkých Newtonových metod je třída po částech hladkých funkcí. Výsledkem je skupina lokálně konvergentních Newtonových metod pro řešení hladkých úloh komplementarity formulovaných jako soustavy po částech hladkých rovnic požívajících funkci minima. Uveďme nejdříve definici po částech hladké funkce.

Definice 5.1. *Spojité funkce $G : \mathcal{D} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$ se nazývá po částech hladká v okolí $x \in \mathcal{D}$, jestliže existuje otevřené okolí $\mathcal{N} \subseteq \mathcal{D}$ bodu x a konečná množina hladkých funkcí $\{G^1, G^2, \dots, G^k\}$ definovaných na \mathcal{N} , pro nějaké kladné celé k , taková, že $G(y)$ je prvek $\{G^1(y), \dots, G^k(y)\}$ pro všechna $y \in \mathcal{N}$. Každá funkce G^i se nazývá hladká část G v x . Nechť $\mathcal{P}(y)$ označuje množinu indexů $i \in \{1, \dots, k\}$ takovou, že $G(y) = G^i(y)$.*

Nechť je tedy $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$ po částech hladká funkce. Předpokládejme, že $x^* \in \mathbb{R}^n$ je nulový bod G . Nechť $\{G^1(x), \dots, G^k(x)\}$ jsou hladké části G v x^* . Přírozené Newtonovo aproximační schéma takové funkce G je dáno jako

$$A(x) \equiv \{JG^i(x) : i \in \mathcal{P}(x)\}.$$

Povšimněte si, že všechny $A(x, \cdot)$ jsou v tomto případě lineární, tj. lineární vzhledem ke druhému argumentu. Neboli jedná se o násobk matice (jakobiánu) a vektoru x . Newtonovy rovnice je proto snadné vyřešit. Navíc učiníme předpoklad, který implikuje regularitu $JG^i(x)$ pro všechna $i \in \mathcal{P}(x)$, takže Newtonovy rovnice budou mít jediné řešení. S tímto aproximačním schématem bude nehladký Newtonův algoritmus v následujícím tvaru.

Algoritmus 5.1 (Po částech hladká Newtonova metoda).

Data: $x^0 \in \mathbb{R}^n$.

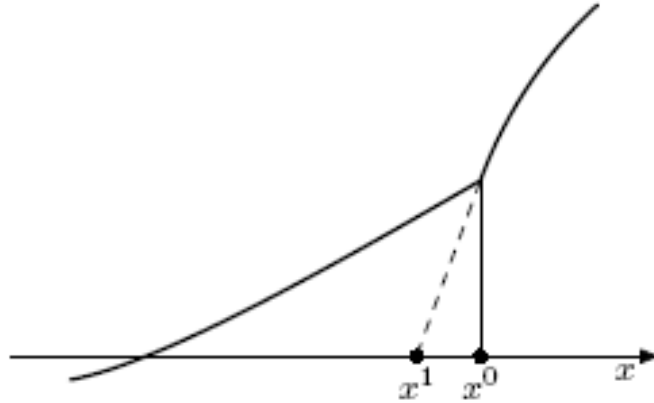
Krok 1: Položíme $k = 0$.

Krok 2: Jestliže $G(x^k) = 0$, stop.

Krok 3: Vybereme index i_k z $\mathcal{P}(x^k)$ a najdeme směr d^k takový, že

$$G(x^k) + JG^{i_k}(x^k)d^k = 0.$$

Krok 4: Položíme $x^{k+1} \equiv x^k + d^k$ a $k \leftarrow k + 1$. Jdeme na krok 2.



Obrázek 4: Ilustrace algoritmu 5.1

Konvergence tohoto algoritmu snadno plyne z věty 4.1. Všimněte si obrázku 4, který ilustruje tento algoritmus.

Věta 5.1. *Nechť $G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$, kde Ω je otevřená, je po částech hladká funkce s hladkými částmi $\{G^i : i = 1 \dots, k\}$. Nechť $x^* \in \Omega$ je nulový bod funkce G . Předpokládejme, že matice $JG^i(x^*)$ jsou regulární pro všechna $i \in \mathcal{P}(x^*)$. Pak existuje okolí $\mathbb{B}(x^*, \delta)$ bodu x^* takové, že pokud x^0 náleží do tohoto okolí, algoritmus 5.1 vytváří jedinou posloupnost $\{x^k\}$, která konverguje k x^* Q -superlineárně. Jestliže jsou jakobiány všech aktivních částí G v bodech blízko x^* lokálně lipschitzovské v x^* , pak je třída konvergence Q -kvadratická.*

Důkaz: Viz [2], str. 657.

Specializací algoritmu 4.2 můžeme snadno získat neexaktní Newtonovu metodu pro funkce po částech hladké. Místo opakování detailů takové neexaktní

metody představíme aplikaci exaktní nehladké Newtonovy metody pro řešení systému rovnic definovaného pomocí funkce minima.

Příklad 5.1. *Mějme nehladkou rovnici $H(x) = 0$, kde*

$$H(x) = \min(F(x), G(x)), \quad \forall x \in \mathbb{R}^n,$$

kde F, G jsou dvě spojitě diferencovatelná zobrazení \mathbb{R}^n do sebe. Nechť je matice $A(x)$ typu $n \times n$ pro libovolný vektor x definovaná řádkově jako

$$A_i \equiv \begin{cases} \nabla F_i(x)^T & \text{pro } F_i(x) < G_i(x), \\ \nabla F_i(x)^T \text{ nebo } \nabla G_i(x)^T & \text{pro } F_i(x) = G_i(x), \\ \nabla G_i(x)^T & \text{pro } F_i(x) > G_i(x). \end{cases}$$

Existuje $2^{|\beta(x)|}$ matic tohoto tvaru pro každé x , kde

$$\beta(x) \equiv \{i : F_i(x) = G_i(x)\}.$$

Je tomu tak proto, že když se pro nějaké i, x hodnota $F_i(x)$ rovná $G_i(x)$, jejich gradienty se rovnat nemusí, a tedy máme dvě možnosti, jak bude i -tý řádek matice A v bodě x vypadat.

Matice A jsou jakobiány hladkých částí funkce H v bodě x . Aplikujeme-li tento poznatek na rovnici $H(x) = 0$, algoritmus 5.1 vytváří posloupnost $\{x^k\}$ následujícím způsobem: pro každé k , kde x^k je dáno, je následující člen x^{k+1} definován jako $x^k + d^k$, kde d^k je řešení systému lineárních rovnic

$$H(x^k) + \tilde{A}(x^k)d^k = 0,$$

kde $\tilde{A}(x^k)$ je jedna z $2^{|\beta(x^k)|}$ matic popsaných výše. Lokální konvergence takové Newtonovy posloupnosti $\{x^k\}$ k nulovému bodu x^ funkce H je zaručena větou 5.1. Abychom tuto větu mohli aplikovat, předpokládáme, že každá matice $A(x^*)$ je regulární.*

Nyní si připomeňme koncept reprezentační matice. Jsou-li A a B dvě matice typu $m \times n$, matice C typu $m \times n$ se nazývá řádková reprezentační matice dvojice (A, B) , jestliže každý řádek C_i je roven buď A_i nebo B_i . Regularita $A(x^)$ je*

proto ekvivalentní podmínce, že každá řádková reprezentační matice M dvojice $(JF(x^*), JG(x^*))$ splňující

$$M_i = \begin{cases} JF_i(x^*) & \forall i \text{ takové, že } F_i(x^*) = 0 < G_i(x^*) \\ JG_i(x^*) & \forall i \text{ takové, že } F_i(x^*) > 0 = G_i(x^*) \end{cases} \quad (5.1)$$

je regulární. Vysvětlení je následující. Předpokládáme-li, že x^* řeší $H(x) = 0$, pak vzhledem k tvaru $H(x)$ je vždy $F_i(x^*)$ nebo $G_i(x^*)$ rovno nule. Druhá funkce je pak nutně kladná, protože jinak by nešlo o nulový bod $H(x)$.

Podotkněme, že u tohoto scénatu leží hlavní otázka každé iterace v řešení systému lineárních rovnic.

Algoritmus naznačený v předchozím příkladě poskytuje lokálně konvergentní metodu pro řešení UK(F, G). Kvůli srozumitelnosti představíme formální popis lokální Newtonovy metody z příkladu 5.1 v pojmech mnohoznačného zobrazení $\mathcal{A}_{min} : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$, kde pro každé $x \in \mathbb{R}^n$ sestává $\mathcal{A}_{min}(x)$ ze všech $n \times n$ matic $A(x)$ definovaných v tomto příkladu.

Algoritmus 5.2 (Newtonova metoda s funkcí minima).

Data: $x^0 \in \mathbb{R}^n$.

Krok 1: Položíme $k = 0$.

Krok 2: Jestliže $\min(F(x^k), G(x^k)) = 0$, stop.

Krok 3: Vybereme $A^k \in \mathcal{A}_{min}(x^k)$ a najdeme řešení d^k lineárního systému

$$\min(F(x^k), G(x^k)) + A^k d = 0. \quad (5.2)$$

Krok 4: Položíme $x^{k+1} = x^k + d^k$ a $k \leftarrow k + 1$. Jdeme na krok 2.

Navíc ke konvergenčním vlastnostem popsaným v příkladě 5.1 je výše uvedený algoritmus skutečně konvergentní v konečném počtu kroků, pokud jsou F, G obě afinní funkce. Pro ucelenost zformulujeme konvergenční vlastnosti algoritmu 5.2 pro nelineární případ a také konečně-konvergenční výsledek pro případ lineární.

Věta 5.2. *Nechť $F, G : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ jsou spojitě diferencovatelné funkce, definované na otevřené množině Ω . Nechť x^* je řešení UK(F, G) takové, že každá řádková reprezentační matice M dvojice $(JF(x^*), JG(x^*))$ splňující (5.1) je regulární. Pak platí následující dvě tvrzení.*

(a) *Existuje okolí $\mathbb{B}(x^0, \delta)$ bodu x^* takové, že pokud x^0 náleží do tohoto okolí, algoritmus 5.2 vytváří jedinou posloupnost $\{x^k\}$, která konverguje k x^* Q -superlineárně. Jestliže jsou JF a JG lokálně lipschitzovské v x^* , pak je třída konvergence Q -kvadratická.*

(b) *Jestliže jsou F, G afinní funkce, pak existuje \bar{k} tak, že $x^{\bar{k}} = x^*$.*

Důkaz: Viz [2], str. 660.

Zabývejme se na závěr této kapitoly jedním speciálním případem rovnice (5.2). Uvažujme případ, kdy G je identické zobrazení. Je-li dáno x^k , zvolíme indexovou množinu $\beta' \subseteq \beta(x^k)$ a řešíme následující systém lineárních rovnic pro d^k :

$$\begin{aligned} F_i(x^k) + \nabla F_i(x^k)^T d^k &= 0 \quad \forall i \in \gamma(x^k) \cup \beta' \\ x_i^k + d_i^k &= 0 \quad \forall i \in \alpha(x^k) \cup (\beta(x^k) \setminus \beta'), \end{aligned}$$

kde

$$\gamma(x^k) \equiv \{i : F_i(x^k) < x_i^k\} \quad \text{a} \quad \alpha(x^k) \equiv \{i : F_i(x^k) > x_i^k\}.$$

O tomto systému platí, že proměnné d_i^k jsou pro $i \in \alpha(x^k) \cup (\beta(x^k) \setminus \beta')$ určeny triviálně, a mohou být dosazeny do první rovnice. Výsledný systém je řádu $|\gamma(x^k) \cup \beta'|$, což může být podstatně méně, než původní dimenze úlohy n . Tato vlastnost je kýžená výhodou algoritmu 5.2 aplikovaného na nelineární úlohu komplementarity, a může být obzvláště zřetelná, pokud počítané řešení x^* obsahuje mnoho nulových prvků.

6 Numerické výpočty

V této rozsáhlé kapitole si stanovíme úlohu deformace nosníku, postupně si odvodíme její různé formulace, představíme si počítačový program a použijeme ho na množství konkrétních příkladů. Začneme popisem základního modelu.

6.1 Formulace úlohy

Zabývejme se Euler-Bernoulliho modelem nosníku

$$(EI u''(x))'' = f(x), \quad x \in (0, L), \quad L > 0. \quad (6.1)$$

Jedná se o eliptickou okrajovou úlohu 4. řádu. Jednotlivé fyzikální veličiny a jejich jednotky jsou uvedeny v následujícím přehledu.

E - Youngův modul pružnosti [Pa]

I - moment setrvačnosti průřezu nosníku vzhledem k pohybové ose [$kg \cdot m^2$]

u - průhyb nosníku [m]

f - vertikální zatížení [N]

L - délka nosníku [m]

Jde o model statický a jednorozměrný, který ale předpokládá, že průřez nosníku je konstantní. Tento průřez se projeví ve veličině I , která tímto způsobem zahrnuje do úlohy další dva rozměry. Dále model předpokládá, že rovinný řez zůstane po deformaci rovinným. Uvažujme nosník, který leží na pevné podložce, a je oboustranně vetknutý, to znamená stabilní, neboli Dirichletovy, okrajové podmínky:

$$\begin{aligned} u(0) &= 0, & u(L) &= 0, \\ u'(0) &= 0, & u'(L) &= 0. \end{aligned}$$

Řešitelností a jednoznačností této úlohy se zabývá např. [8], str. 230.

6.2 Diskretizace úlohy

Na rovnici nosníku použijeme Galerkinovu metodu, a postupně získáme formulaci úlohy ve tvaru minimalizace funkcionálu, příslušnou diskretizovanou úlohu a nakonec maticovou rovnost.

Označme

$$V = H_0^2(0, L).$$

Nejdříve všechny členy rovnice (6.1) převedeme na jednu stranu. Celou rovnici přenásobíme testovací funkcí $v \in V$, libovolnou pevnou, a zintegrujeme v mezích od 0 do L podle proměnné x . Máme

$$\int_0^L (EI u''(x))'' v(x) dx - \int_0^L f(x)v(x) dx = 0.$$

Na první člen rovnice dvakrát aplikujeme integraci per partes:

$$\begin{aligned} [(EI u'')'v]_0^L - \int_0^L (EI u''(x))'v'(x) dx - \int_0^L f(x)v(x) dx &= 0, \\ [(EI u'')'v]_0^L - [EI u''v']_0^L + \int_0^L EI u''(x)v''(x) dx - \int_0^L f(x)v(x) dx &= 0. \end{aligned}$$

Využijeme vlastností testovací funkce v , díky kterým jsou první dva výrazy v poslední rovnici rovny nule, a dostáváme slabou formulaci

$$\int_0^L EI u''(x)v''(x) dx - \int_0^L f(x)v(x) dx = 0, \quad \forall v \in V. \quad (6.2)$$

Označme J funkcionál, jehož G-diferenciál je v bodě u ve směru v roven výrazu na levé straně slabé formulace (6.2). Tento funkcionál tvaru

$$J(v) = \frac{1}{2} \int_0^L EI (v'')^2 dx - \int_0^L f v dx, \quad v \in V,$$

se nazývá funkcionál potenciální energie nosníku, a jeho členy odpovídají potenciální energii vnitřních a vnějších sil. Kvůli jednoznačnosti a řešitelnosti úlohy

potřebujeme ověřit konvexitu tohoto funkcionálu. Podívejme se na jeho druhý diferenciál:

$$J''(u, v, v) = \int_0^L EI(v'')^2 dx, \quad u, v \in V.$$

Vidíme, že je vždy nezáporný, protože E, I jsou kladné funkce. To znamená, že J je vskutku konvexní. Ve skutečnosti je dokonce ryze konvexní, což vplyne z vyšetřování V -elipticity o několik řádků níže. Protože nosník leží na pevné podložce, označme jako K množinu

$$K = \{v \in H_0^2(0, L) : v(x) \geq 0, \forall x \in (0, L)\}.$$

K je konvexní uzavřená podmnožina prostoru $H_0^2(0, L)$. Nyní si můžeme uvést druhý způsob formulace úlohy: naším úkolem je minimalizovat potenciální energii nosníku, tedy najít $u \in K$ takové, že

$$J(u) = \min_{v \in K} J(v). \quad (6.3)$$

Na základě poznatků z variačních metod platí, že konvexní a G -diferencovatelný funkcionál J nabývá svého minima v bodě $u \in K$, kde K je neprázdná konvexní množina, právě když

$$J'(u, v - u) \geq 0, \quad \forall v \in K. \quad (6.4)$$

Označíme-li

$$a(u, v) = \int_0^L EIu''v'' dx, \quad u, v \in V,$$

s použitím nerovnosti (6.4) lze psát úlohu (6.3) ve tvaru: najít $u \in K$ takové, že

$$a(u, v - u) \geq \langle f, v - u \rangle, \quad \forall v \in K, \quad (6.5)$$

kde $\langle \cdot, \cdot \rangle : V^* \times V \rightarrow \mathbb{R}$ je dualita. Tato formulace je ekvivalentní s minimalizací funkcionálu J právě díky tomu, že J je konvexní. Směřujme nyní k větě o existenci

a jednoznačnosti řešení úlohy (6.5). Následující úvahy provedeme pro E, I konstantní, ale lze je zobecnit i pro předpoklad, že E, I jsou kladné funkce. Vezměme libovolnou funkci $v \in V$, pak platí $v \in H_0^1$, $v' \in H_0^1$, a s použitím Friedrichsovy nerovnosti můžeme psát

$$\begin{aligned} \|v\|_{H^1}^2 &= \int_0^L (v^2 + (v')^2) dx \leq c_0 \int_0^L (v')^2 dx, \\ \|v'\|_{H^1}^2 &= \int_0^L ((v')^2 + (v'')^2) dx \leq c_1 \int_0^L (v'')^2 dx. \end{aligned}$$

Dále platí

$$\begin{aligned} a(v, v) &= \int_0^L EI(v'')^2 dx \geq \frac{EI}{c_1} \int_0^L ((v')^2 + (v'')^2) dx \geq \\ &\geq \frac{EI}{c_1 c_0} \int_0^L (v^2 + (v')^2) dx + \frac{EI}{c_1} \int_0^L (v'')^2 dx \geq \frac{EI}{c_1 \max(c_0, 1)} \|v\|_V^2. \end{aligned}$$

Protože $v \in V$ bylo libovolné, dokázali jsme, že forma a je V -eliptická, a můžeme použít následující větu.

Věta 6.1. *Nechť $a : V \times V \rightarrow \mathbb{R}$ je V -eliptická, symetrická bilineární forma, omezená na V , a nechť $f \in V^*$. Potom existuje právě jediné řešení úlohy (6.5).*

Důkaz: Viz [5], str. 15.

Nyní si v souladu s metodou konečných prvků rozdělíme interval $[0, L]$ na n prvků, přičemž krajními body každého prvku budou dva uzly. Uzly si označme x_0, x_1, \dots, x_n , je jich tedy celkem $n+1$. Rozmístíme je ekvidistantně na intervalu $[0, L]$, s tím, že uzel x_0 splývá s bodem 0, a x_n s bodem L . Budeme požadovat, aby posunutí i jeho derivace byly při přechodu mezi sousedními prvky spojitě. Abychom tuto podmínku splnili, v každém uzlu budeme uvažovat posunutí a jeho derivaci jako uzlové proměnné. To znamená, že sousední prvky mají stejnou hodnotu posunutí a derivace ve společném uzlu, a tedy podmínka spojitosti je

splněna. Jako další krok diskretizujeme prostor V a množinu K a označme

$$\begin{aligned} V_h &= \{v_h \in C^1([0, L]) : v_h|_{[x_i, x_{i+1}]} \in P_3([x_i, x_{i+1}]), \forall i = 0, \dots, n, \\ &\quad v_h(0) = 0, v_h'(0) = 0, v_h(L) = 0, v_h'(L) = 0\}, \\ K_h &= \{v_h \in V_h : v_h(x_i) \geq 0, \forall i = 1, \dots, n-1\}. \end{aligned}$$

Je známo, že platí

$$V_h \subset V,$$

ale nepožadujeme $K_h \subset K$. V tuto chvíli můžeme zformulovat diskretizovanou úlohu: hledáme $u_h \in K_h$ takové, že splňuje

$$a(u_h, v_h - u_h) \geq \langle f, u_h - v_h \rangle, \quad \forall v_h \in K_h. \quad (6.6)$$

Vezmeme-li V_h za V a K_h za K , věta 6.1 nám říká, že i tato úloha má právě jediné řešení. Vztah mezi těmito úlohami řeší věta následující.

Věta 6.2. *S výše uvedenými předpoklady platí*

$$\lim_{h \rightarrow 0} \|u_h - u\|_V = 0,$$

kde u_h je řešení úlohy (6.6), a u řešení (6.5).

Důkaz: Viz [4], str. 10.

Hodnoty posunutí u a jeho derivace v uzlech x_0, x_1, \dots, x_n si seřadíme do vektoru, a dále proto budeme pracovat s vektory délky $2(n+1)$ následujícího tvaru:

$$\mathbf{u} = (u_h(x_0), u_h'(x_0), u_h(x_1), u_h'(x_1), \dots, u_h(x_n), u_h'(x_n)).$$

Označme si

$$\mathcal{K} = \{\mathbf{v} \in \mathbb{R}^{2(n+1)} : \mathbf{v}_j \geq 0 \text{ pro } j \text{ liché}\}.$$

Další postup metody konečných prvků využívá hermiteovských polynomů 3. stupně a vede k získání matice tuhosti \mathbf{M} a vektoru zatížení \mathbf{F} . S jejich pomocí lze

zapsat konečně dimenzonální aproximaci diskretizované úlohy: hledáme $\mathbf{u} \in \mathcal{K}$ tak, že

$$(\mathbf{M}\mathbf{u}, \mathbf{u} - \mathbf{v}) \geq (\mathbf{F}, \mathbf{u} - \mathbf{v}), \quad \forall \mathbf{v} \in \mathcal{K}.$$

Na základě této formulace v naší úloze dále platí

$$\begin{aligned} \mathbf{u}_j &\geq 0 && \text{pro } j \text{ liché,} \\ (\mathbf{M}\mathbf{u} - \mathbf{F})_j &\geq 0 && \text{pro } j \text{ liché,} \\ (\mathbf{M}\mathbf{u} - \mathbf{F})_j &= 0 && \text{pro } j \text{ sudé,} \end{aligned}$$

přičemž z rovnic pro j liché je vždy právě jedna větší rovna nule a druhá rovna nule pro každé j . Tyto vztahy můžeme přepsat na

$$\begin{aligned} \min\{\mathbf{u}_j, (\mathbf{M}\mathbf{u} - \mathbf{F})_j\} &= 0 && \text{pro } j \text{ liché,} \\ (\mathbf{M}\mathbf{u} - \mathbf{F})_j &= 0 && \text{pro } j \text{ sudé.} \end{aligned}$$

Získali jsme soustavu nehladkých rovnic, na niž lze použít algoritmus 5.2.

Zabývejme se nyní obecnějším případem, kdy pevná podložka může být níže než nosník, ve tvaru libovolné hladké nekladné funkce $\varphi \in C^1([0, L])$. Tuto regularitu požadujeme proto, že pro výpočet je třeba znát derivaci funkce φ . Podívejme se, jaké změny je třeba provést v postupu uvedeném výše. Jednotlivé kroky diskretizace jsou stejné, ovšem uvažujeme jinou množinu, na které minimalizujeme funkcional J . Označme

$$K^\varphi = \{v \in H_0^2(0, L) : v(x) \geq \varphi(x), \forall x \in (0, L)\}.$$

K^φ je opět konvexní uzavřená podmnožina prostoru $H_0^2(0, L)$. Úloha minimalizace potenciální energie nosníku má následující tvar: najít $u \in K^\varphi$ takové, že

$$J(u) = \min_{v \in K^\varphi} J(v).$$

Protože i množina K^φ má požadované vlastnosti, lze psát, že funkcional potenciální energie J nabývá svého minima v bodě $u \in K^\varphi$, právě když

$$a(u, v - u) \geq \langle f, v - u \rangle, \quad \forall v \in K^\varphi.$$

Diskretizace množiny K^φ je následující:

$$K_h^\varphi = \{v_h \in V_h : v_h(x_i) \geq \varphi(x_i), \forall i = 1, \dots, n-1\}.$$

Stejně jako v předchozím jsme dospěli k úloze najít $u_h \in K_h^\varphi$ takové, že splňuje

$$a(u_h, v_h - u_h) \geq \langle f, u_h - v_h \rangle, \quad \forall v_h \in K_h^\varphi.$$

Další postup podle metody konečných prvků je opět analogický, a označením

$$\mathcal{K}^\varphi = \{\mathbf{v} \in \mathbb{R}^{2(n+1)} : \mathbf{v}_j \geq \varphi(x_{(j-1)/2}) \text{ pro } j \text{ liché}\}$$

můžeme zapsat následující úlohu: najít $\mathbf{u} \in \mathcal{K}^\varphi$ tak, že

$$(\mathbf{M}\mathbf{u}, \mathbf{u} - \mathbf{v}) \geq (\mathbf{F}, \mathbf{u} - \mathbf{v}), \quad \forall \mathbf{v} \in \mathcal{K}^\varphi.$$

Platí

$$\begin{aligned} \mathbf{u}_j &\geq \varphi(x_{(j-1)/2}) && \text{pro } j \text{ liché,} \\ (\mathbf{M}\mathbf{u} - \mathbf{F})_j &\geq 0 && \text{pro } j \text{ liché,} \\ (\mathbf{M}\mathbf{u} - \mathbf{F})_j &= 0 && \text{pro } j \text{ sudé,} \end{aligned}$$

přičemž z rovnic pro j liché je vždy právě jedna větší rovna nule a druhá rovna nule pro každé j . Dále lze psát

$$\begin{aligned} \min\{\mathbf{u}_j - \varphi(x_{(j-1)/2}), (\mathbf{M}\mathbf{u} - \mathbf{F})_j\} &= 0 && \text{pro } j \text{ liché,} \\ (\mathbf{M}\mathbf{u} - \mathbf{F})_j &= 0 && \text{pro } j \text{ sudé.} \end{aligned}$$

Opět jsme získali soustavu nehladkých rovnic, na které lze aplikovat Newtonovu metodu s funkcí minima.

6.3 Počítačový program

Pro výpočet výše popsané úlohy prohnutí nosníku s použitím algoritmu 5.2 jsem vytvořila v softwaru MATLAB R2008a M-file `newton_min`, a dále M-fily `kontrola_vstupu` a `zpracovani_vstupu`, které program `newton_min` volá. Pro sestavení matice tuhosti M a vektoru zatížení F používám form `StiffnessBernoulli-Beam_mod`, který vznikl upravením M-filu ze [3], str. 84. Přestože se předchozí úvahy zabývají pouze oboustranně vetknutým nosníkem, program dokáže pracovat i s nosníkem prostě podepřeným, přičemž počet, místa a typy uchycení jsou volitelné. Uvedené programy jsou k dispozici na příloženém CD.

Vstup a výstup programu `newton_min` jsem podrobně popsala v samotném kódu. Je třeba zadat charakteristiky nosníku, tj. délku, Youngův modul pružnosti E a moment setrvačnosti I , a také jeho uchycení, tedy je-li nosník vetknut či prostě podepřen, a ve kterých bodech. Lze zadat vnější síly ve formě zatížení, osamělých sil, případně jejich kombinace, a také pevnou podložku. Dále je třeba zvolit počet prvků, na něž nosník rozdělíme, přesnost výpočtu, počáteční bod Newtonovy metody, a rozhodovací parametr. Tento parametr se použije v případě rovnosti dvou funkcí, z nichž určujeme minimum.

Výstupem je vektor uzlů, vektor hodnot posunutí a jeho derivací, počet iterací Newtonovy metody, body největší výchylky v absolutní hodnotě, a hodnota výchylky. Grafický výstup zobrazuje vypočítané posunutí a zadanou pevnou podložku. Na druhém grafu lze vidět derivaci posunutí a derivaci funkce, která určuje tvar podložky. Protože jsou grafy černobílé, pro přehlednost je použito různých druhů značek. Barevné provedení lze nalézt v elektronické formě na příloženém CD.

M-file newton_min

```
% VSTUP
%%% v zavorkach je vzdy uveden rozmer dane promenne
% L - delka nosniku [m] - (1,1)
% n - pocet prvku - (1,1)
% uchyceni - obsahuje body a zpusob uchyceni - (pocet uchyceni,2)
%           - misto uchyceni se zadava pomerne
%           (bez ohledu na delku nosniku) jako cislo od 0 do 1
%           - syntaxe jednoho radku: [misto, zpusob]
%           proste podepren: zpusob = 0
%           vetknut: zpusob = 1
% E - Younguv modul pruznosti [Pa] - (1,1)
% Izlom,Ihodn - odpovidaji I, coz je moment setrvacnosti
%           prurezu nosniku vzhledem k pohybove ose [kg*m^2]
%           - I uvazujeme po castech konstantni:
%           Izlom - mista zmeny hodnoty I
%                   - (0,0) nebo (pocet zlomu,1)
%                   - misto se zadava jako cislo od 0 do 1
%                   - je-li I vsude stejny, zada se []
%           Ihodn - hodnoty I mezi zlomy
%                   - (1,1) nebo (pocet zlomu+1,1)
%                   - vzdy jde o vektor o 1 delsi nez Izlom
%                   - priklad: Izlom=1/2, Ihodn=[1,10]
% p - zatizeni (vnejsi sila) [N] - (funkce)
%   - pouzity budou funkcní hodnoty na intervalu [0,L]
%   - zaporne zatizeni pusobi smerem dolu
% f - osamele sily (vnejsi sila) [N] - (0,0) nebo (pocet sil,2)
%   - syntaxe jednoho radku: [misto pusobeni, velikost sily]
%   - misto se zadava jako cislo od 0 do 1
%   - zaporna sila pusobi smerem dolu
```

```

% - nepusobi-li zadna sila, zada se prazdny vektor []
% u0 - pocatecni odhad reseni - (2*(n+1),1)
% epsilon - pozadovana presnost (norma rezidua soustavy musi byt
%      mensi nez epsilon) - (1,1)
% parametr - logicka hodnota urcujici tvar gradientu funkce
%      min(x,M*x-f) v pripade rovnosti funkci x a M*x-f:
%      true - uvazujeme gradient funkce x
%      false - uvazujeme gradient funkce M*x-f
% fi - pevna podlozka nachazejici se pod nosnikem (funkce)
% - funkcní hodnoty musí být nekladne
% - pouzity budou funkcní hodnoty na intervalu [0,L]
% dfi - derivace funkce fi (funkce)
%
% VYSTUP
% x - vektor uzlu - (n+1,1)
% U - vektor posunuti a jeho derivaci v uzlech - (2*(n+1),1)
%      U = [ u(x0),u'(x0),u(x1),u'(x1),...,u(xn),u'(xn) ]
% k - pocet iteraci Newtonovy metody - (1,1)
% xmax - body maximalni vychylky, udano cislem od 0 do 1
%      - (poc. bodu,1)
% umax - hodnota maximalni vychylky - (1,1)

function [x,U,k,xmax,Umax] = newton_min (L,n,uchyceni,E, ...
      Izlom,Ihodn,p,f,u0,epsilon,parametr,fi,dfi)

% --- kontrola a zpracovani vstupu

kontrola_vstupu(L,n,uchyceni,E,Izlom,Ihodn,f,u0,epsilon,parametr);
[uzly,predepsano,I,P,G,Fi,Dfi] = ...
      zpracovani_vstupu(L,n,uchyceni,Izlom,Ihodn,p,f,fi,dfi);

```

```

% --- ziskani matice a vektoru soustavy, inicializace

[M,F] = formStiffnessBernoulliBeam_mod(n,uzly,E,I,P);
aktivni = setdiff((1:2*(n+1))',predepsano); % *solution.m
akti = setdiff((1:2*(n+1))',[predepsano;(2:2:2*(n+1))']);
akti = (akti+1)/2;
M = M(aktivni,aktivni); % *solution.m
F = F+G;
F = F(aktivni); % *solution.m
m = length(F);
Fi = Fi(akti);
Dfi = Dfi(akti);
u = u0(aktivni);
k = 0;

% --- nulta iterace

A = M;
H = M*u-F;
jj = -1;
p = 1;

for j=1:length(akti)
    jj = jj+2;
    if (predepsano(p)+1)==aktivni(jj)
        jj = jj+1;
        p = p+1;
    end
    if (u(jj)-Fi(j))==H(jj) && parametr==true
        disp('Nastala rovnost.')
```

```

        A(jj,:) = zeros(1,m);
        A(jj,jj) = 1-Dfi(j);
elseif (u(jj)-Fi(j))<H(jj)
        A(jj,:) = zeros(1,m);
        A(jj,jj) = 1-Dfi(j);
        H(jj) = u(jj)-Fi(j);
    end
end

% --- cyklus s podminkou

while norm(H) > epsilon && k<100

    d = A\(-H);
    u = u+d;
    A = M;
    H = M*u-F;
    jj = -1;
    p = 1;

    for j=1:length(akti)
        jj = jj+2;
        if (predepsano(p)+1)==aktivni(jj)
            jj = jj+1;
            p = p+1;
        end
        if (u(jj)-Fi(j))==H(jj) && parametr==true
            disp('Nastala rovnost.')
            A(jj,:) = zeros(1,m);
            A(jj,jj) = 1-Dfi(j);
        end
    end
end

```



```

elseif (u(jj)-Fi(j))<H(jj)
    A(jj,:) = zeros(1,m);
    A(jj,jj) = 1-Dfi(j);
    H(jj) = u(jj)-Fi(j);
end
end

k = k+1;

end

% --- graficke znazorneni posunuti, vypis poctu iteraci

U=zeros(2*(n+1),1); % *solution.m
U(aktivni)=u; % *solution.m
posunuti = U(1:2:(2*n+1));
derivace = U(2:2:2*(n+1));
[Uabs,kde] = max(abs(posunuti));
Umax = posunuti(kde);
pril = abs(posunuti-Umax*ones(n+1,1))<epsilon;
pril = nonzeros(pril.*(1:(n+1))');
xmax = uzly(pril)/L;
x = uzly;

subplot(2,1,1)
plot(uzly,posunuti,'.')
legend('fi','u','Location','SouthEast')
stred1 = (max(posunuti)+min(posunuti))/2;
delka1 = abs(max(posunuti)-min(posunuti))/2;
axis([0,L,stred1-1.2*delka1,stred1+1.2*delka1])

```

```

subplot(2,1,2)
plot(uzly,derivace,'.')
legend('fi','','u','','Location','SouthEast')
stred2 = (max(derivace)+min(derivace))/2;
delka2 = abs(max(derivace)-min(derivace))/2;
axis([0,L,stred2-1.2*delka2,stred2+1.2*delka2])

sprintf('Pocet iteraci: %g \nBody maximalni vychylky: [%g,%g] \n
Hodnota maximalni vychylky: %0.4e',k,xmax(1),xmax(end),Umax)

% VYSVETLIVKY
% *solution.m - prevzato ze solution.m a upraveno

```

M-file kontrola_vstupu

```

function kontrola_vstupu (L,n,uchyceni,E,Izlom,Ihodn,f,u0, ...
    epsilon,parametr)

Dzlom = length(Izlom);
Dhodn = length(Ihodn);

if L<=0
    disp('Chyba na vstupu: L musi byt kladne cislo.')
elseif n<=0
    disp('Chyba na vstupu: n musi byt kladne cislo.')
elseif isequal(n,round(n))==0
    disp('Chyba na vstupu: n musi byt cele cislo.')
elseif sum(size(uchyceni)<[1,2])>0
    disp('Chyba na vstupu: nosnik musi byt uchycen
alespon v 1 bode.')
elseif (sum(uchyceni(:,1)<0)>0) || (sum(uchyceni(:,1)>1)>0)

```

```

disp('Chyba na vstupu: uchyceni musi mit v 1. sloupci
hodnoty z intervalu [0,1].')
elseif sum((uchyceni(:,2)~=0)+(uchyceni(:,2)~=1))~=size(uchyceni,1)
disp('Chyba na vstupu: uchyceni musi mit v 2. sloupci
hodnoty 0 nebo 1.')
```

```
elseif E<=0
disp('Chyba na vstupu: E musi byt kladne cislo.')
```

```
elseif (Dzlom+1)~=Dhodn
disp('Chyba na vstupu: Ihodn musi byt o 1 delsi vektor
nez Izlom.')
```

```
elseif Dhodn>0 && sum(Ihodn>0)==0
disp('Chyba na vstupu: Ihodn musi byt vektor kladnych cisel.')
```

```
elseif Dzlom>0 && ( (sum(Izlom<0)>0) || (sum(Izlom>1)>0) )
disp('Chyba na vstupu: Izlom musi obsahovat hodnoty
z intervalu [0,1].')
```

```
elseif (sum(size(f))>0) && (size(f,2)~=2)
disp('Chyba na vstupu: f musi mit 2 sloupce,');
disp('nebo musi byt prazdny vektor [].')
```

```
elseif (sum(size(f))>0) && ( (sum(f(:,1)<0)>0) || (sum(f(:,1)>1)>0) )
disp('Chyba na vstupu: f musi mit v 1. sloupci
hodnoty z intervalu [0,1].')
```

```
elseif size(u0)~= [2*(n+1),1]
disp('Chyba na vstupu: u0 musi byt sloupcovy vektor
delky 2*(n+1).')
```

```
elseif epsilon<=0
disp('Chyba na vstupu: epsilon musi byt kladne cislo.')
```

```
elseif islogical(parametr)==0
disp('Chyba na vstupu: parametr musi byt logicka hodnota.')
```

```
end
```

M-file zpracovani_vstupu

```
function [uzly,predepsano,I,P,G,Fi,Dfi] = ...
    zpracovani_vstupu(L,n,uchyceni,Izlom,Ihodn,p,f,fi,dfi)

tridit = uchyceni(:,1);
if sum(size(Izlom))>0
    tridit = [tridit;Izlom];
end
if sum(size(f))>0
    tridit = [tridit;f(:,1)];
end
zlomy = sort(tridit);
if zlomy(1)~=0
    zlomy = [0;zlomy];
end
if zlomy(end)~=1
    zlomy = [zlomy;1];
end

pomery = round((n+1)*zlomy);
pomery(2:end) = pomery(2:end)-1;
pomery(end) = n;

uzly = 0;
zlomuzl = [1,0];
for s=1:(length(pomery)-1)
    dalsi = linspace(zlomy(s),zlomy(s+1),pomery(s+1)-pomery(s)+1)';
    uzly = [uzly; dalsi(2:end)];
    zlomuzl = [zlomuzl;length(uzly),uzly(end)];
end
```

```

uzly = L*uzly;
dz = size(zlomuzl,1);

% pro nasledujici vypocet predepsano pouzita myslenka z problem9.m
duch = size(uchyceni,1);
predepsano = [];
for o=1:duch
    for oo=1:dz
        if uchyceni(o,1)==zlomuzl(oo,2)
            predepsano = [predepsano;2*zlomuzl(oo,1)-1];
            if uchyceni(o,2)==1
                predepsano = [predepsano;2*zlomuzl(oo,1)];
            end
        end
    end
end
end
P = p(uzly);
if length(P)==1
    P = P*ones(n+1,1);
end

df = size(f,1);
G = zeros(2*(n+1),1);
for q=1:df
    for qq=1:dz
        if f(q,1)==zlomuzl(qq,2)
            G(2*zlomuzl(qq,1)-1) = f(q,2);
        end
    end
end
end
end

```

```

I = [];
Iuzel = 1;
for r=1:length(Izlom)
    for rr=1:dz
        if Izlom(r)==zlomuzl(rr,2)
            I = [I;Ihodn(r).*ones(zlomuzl(rr,1)-Iuzel,1)];
            Iuzel = zlomuzl(rr,1);
        end
    end
end
end
I = [I;Ihodn(end).*ones(n+1-Iuzel,1)];

Fi = fi(uzly);
if length(Fi)==1
    Fi = Fi*ones(n+1,1);
end
if (sum(Fi>zeros(n+1,1)))>0
    disp('Chyba na vstupu: funkce fi musi byt nekladna.')
end

Dfi = dfi(uzly);
if length(Dfi)==1
    Dfi = Dfi*ones(n+1,1);
end

subplot(2,1,1)
plot(uzly,Fi,'--r')
hold on;
subplot(2,1,2)
plot(uzly,Dfi,'r--')

```

```
hold on;
```

```
% VYSVETLIVKY
```

```
% *solution.m - prevzato ze solution.m a upraveno
```

M-file formStiffnessBernoulliBeam_mod

```
function [stiffness,force]=...
```

```
    formStiffnessBernoulliBeam_mod(numberElements,xx,E,I,P); % upr.
```

```
GDof=2*(numberElements+1); % pridano
```

```
elementNodes=[(1:numberElements)',(2:numberElements+1)']; % pridano
```

```
force=zeros(GDof,1);
```

```
stiffness=zeros(GDof);
```

```
% calculation of the system stiffness matrix
```

```
% and force vector
```

```
for e=1:numberElements;
```

```
    % elementDof: element degrees of freedom (Dof)
```

```
    indice=elementNodes(e,:);
```

```
    elementDof=[ 2*(indice(1)-1)+1 2*(indice(2)-1)...
```

```
                2*(indice(2)-1)+1 2*(indice(2)-1)+2];
```

```
    % ndof=length(indice);
```

```
    % length of element
```

```
    LElem=xx(indice(2))-xx(indice(1));
```

```
    %ll=LElem; % zapoznamkovano
```

```
    EI=E*I(e); % pridano
```

```
    k1=EI/(LElem)^3*[12 6*LElem -12 6*LElem;
```

```
                    6*LElem 4*LElem^2 -6*LElem 2*LElem^2;
```

```
                    -12 -6*LElem 12 -6*LElem ;
```

```
                    6*LElem 2*LElem^2 -6*LElem 4*LElem^2];
```

```

%f1=[P*LElem/2 P*LElem*LElem/12 P*LElem/2 ... % zapoznamkovano
%   -P*LElem*LElem/12]'; % zapoznamkovano
T=P([indice(1) indice(2)]); % pridano
f1=[T(1)*LElem/2 T(1)*LElem*LElem/12 T(2)*LElem/2 ...
    -T(2)*LElem*LElem/12]'; % upraveno

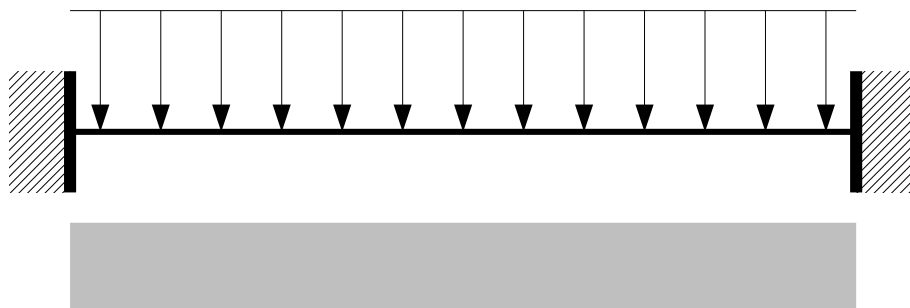
% equivalent force vector
force(elementDof)=force(elementDof)+f1;
% stiffness matrix
stiffness(elementDof,elementDof)=...
    stiffness(elementDof,elementDof)+k1;
end

```


6.4 Příklady

Právě představený počítačový program jsem použila k výpočtu příkladů, přičemž příklady 6.8, 6.9 jsem dostala zadány, a zbylé jsem vytvářela s ohledem na co největší názornost. Každý příklad obsahuje popis, který vystihuje situaci, schematický náčrt a tabulku zadávaných fyzikálních veličin. V souladu s programem značím zatížení p . Jako počáteční bod beru vždy nulový vektor, a parametr je vždy nastaven na hodnotu false. Pokud se vyskytují osamělé vnější síly, jsou uvedeny ve speciální tabulce. U každého příkladu jsem provedla výpočet pro 30, 50 a 80 prvků. V tabulce výsledků zaznamenávám počet prvků, počet iterací, body maximální výchylky a její hodnotu. Výsledky pro počet prvků 50 zobrazuji pomocí grafů.

Příklad 6.1 (Oboustranně vetknutý nosník, rovnoměrné zatížení, vodorovná podložka níže než nosník).

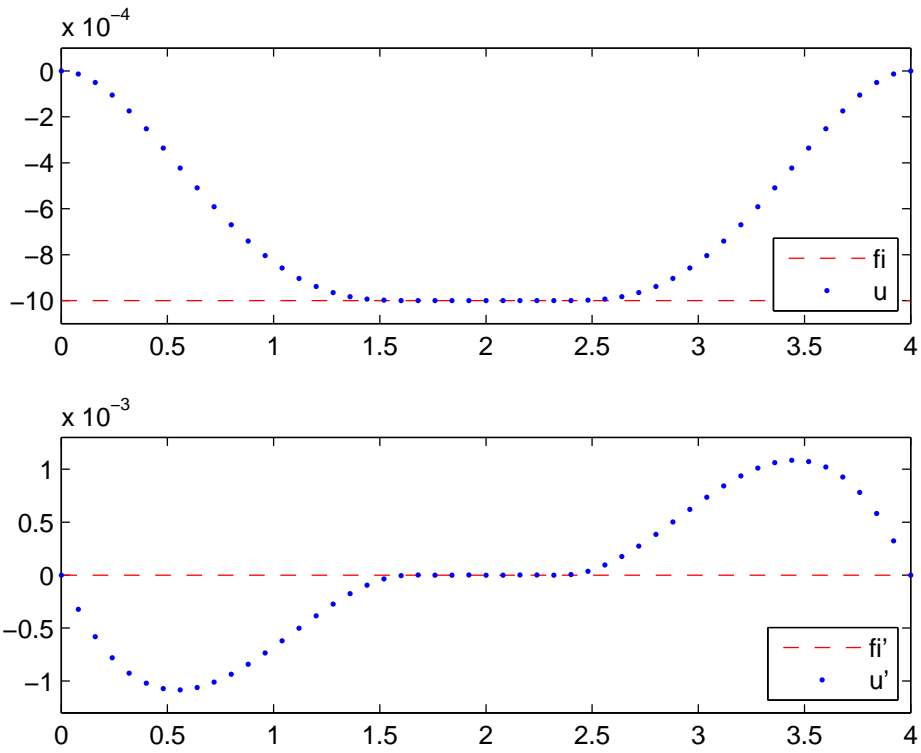


Zadání:

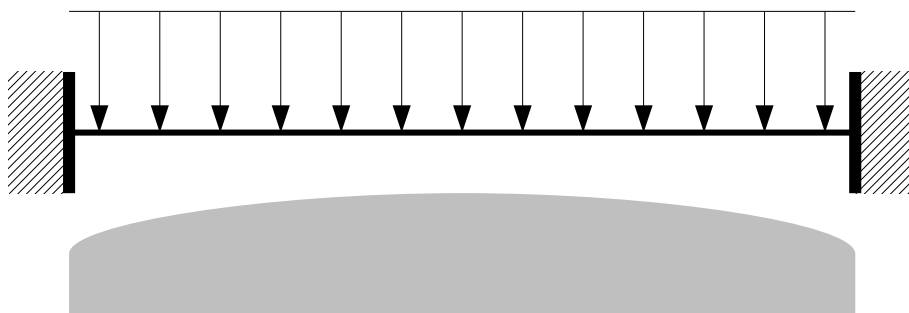
L	E	I	p	ε	φ
4	100	1	-1	10^{-7}	-10^{-3}

Výsledky:

<i>Počet prvků</i>	30	50	80
<i>Počet iterací</i>	14	22	34
<i>Body maximální výchylky</i>	[0, 4; 0, 6]	[0, 4; 0, 6]	[0, 4; 0, 6]
<i>Hodnota maximální výchylky</i>	-10^{-3}	-10^{-3}	-10^{-3}

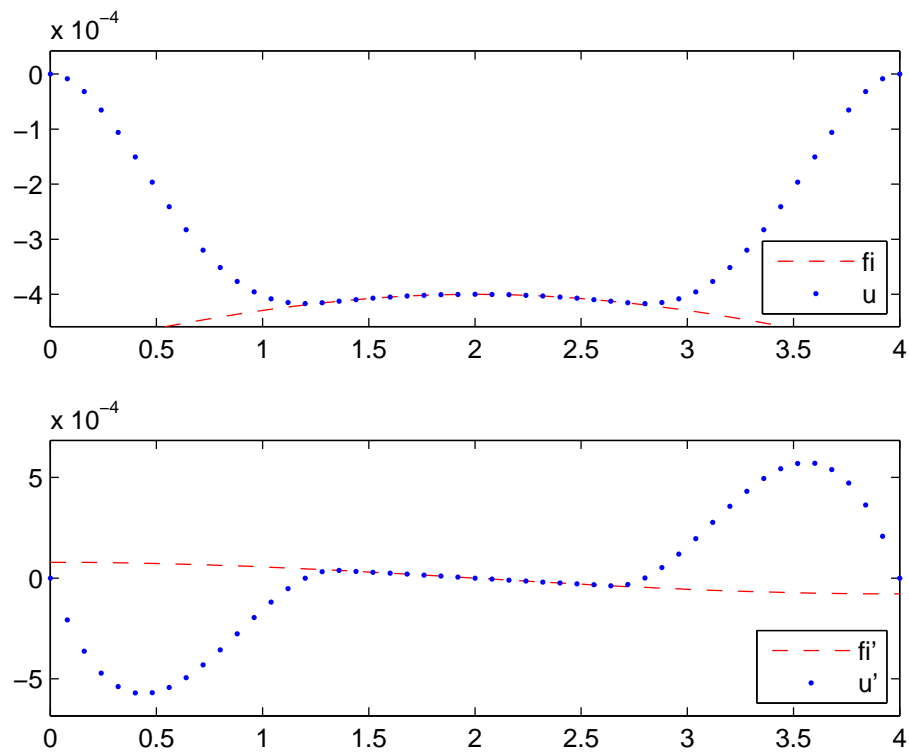


Příklad 6.2 (Oboustranně vetknutý nosník, rovnoměrné zatížení, podložka tvaru sinusovky níže než nosník).



Zadání:

$$\begin{array}{cccccc}
 L & E & I & p & \varepsilon & \varphi \\
 4 & 100 & 1 & -1 & 10^{-7} & 10^{-4} \sin\left(\frac{\pi}{L}x\right) - 5 \cdot 10^{-4}
 \end{array}$$



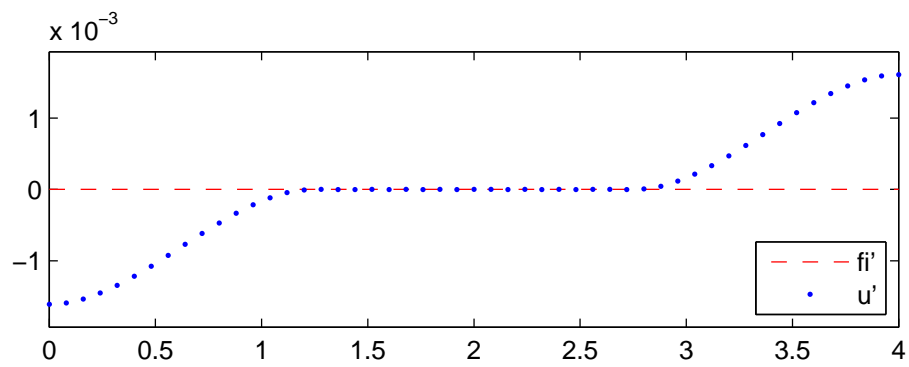
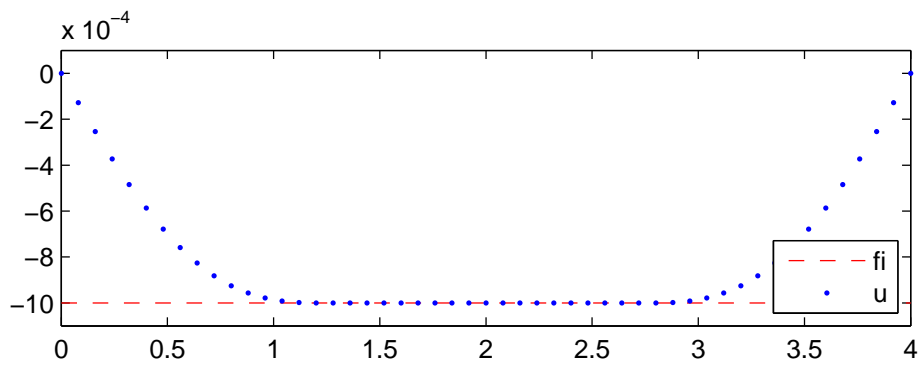
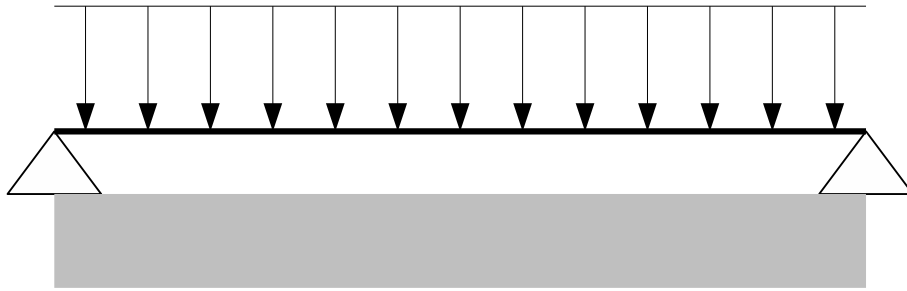
Výsledky:

Počet prvků	30	50	80
Počet iterací	10	18	28
Body maximální výchylky	$[0, 3; 0, 7]$	$[0, 3; 0, 7]$	$[0, 3; 0, 7]$
Hodnota maximální výchylky	$-4.1636 \cdot 10^{-4}$	$-4.1673 \cdot 10^{-4}$	$-4.1659 \cdot 10^{-4}$

Příklad 6.3 (Na krajích prostě podepřený nosník, rovnoměrné zatížení, vodorovná podložka níže než nosník).

Zadání:

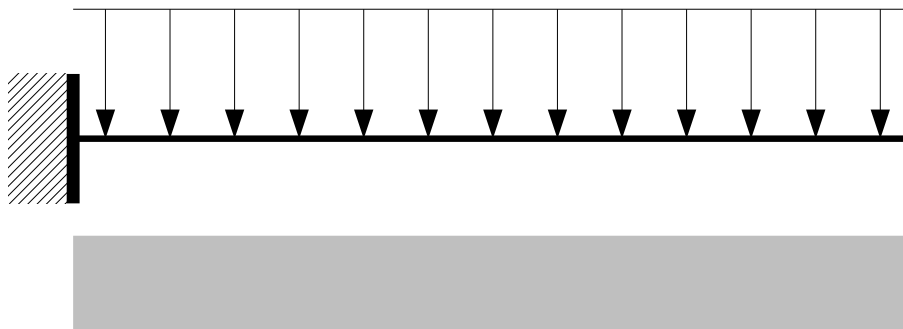
L	E	I	p	ε	φ
4	100	1	-1	10^{-7}	-10^{-3}



Výsledky:

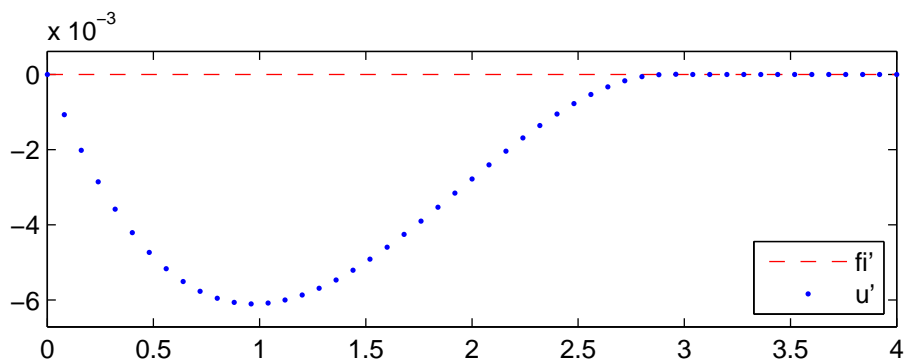
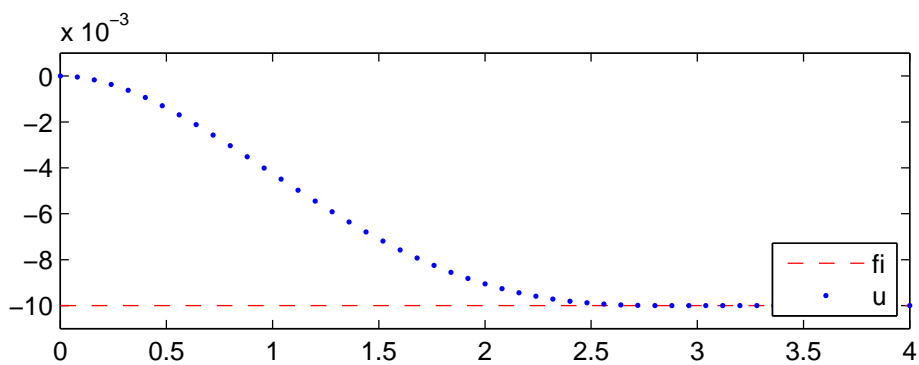
Počet prvků	30	50	80
Počet iterací	12	15	26
Body maximální výchylky	[0, 3; 0, 7]	[0, 3; 0, 7]	[0, 3; 0, 7]
Hodnota maximální výchylky	-10^{-3}	-10^{-3}	-10^{-3}

Příklad 6.4 (Jednostranně vetknutý nosník, rovnoměrné zatížení, vodorovná podložka níže než nosník).



Zadání:

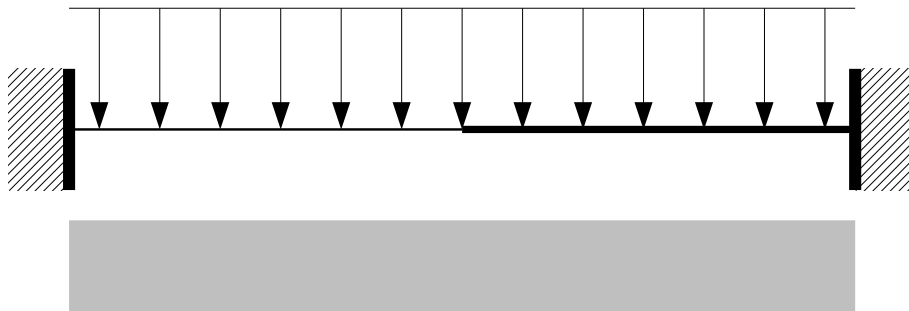
L	E	I	p	ε	φ
4	100	1	-1	10^{-7}	-10^{-2}



Výsledky:

Počet prvků	30	50	80
Počet iterací	24	38	61
Body maximální výchylky	[0, 733333; 1]	[0, 72; 1]	[0, 725; 1]
Hodnota maximální výchylky	-10^{-2}	-10^{-2}	-10^{-2}

Příklad 6.5 (Oboustranně vetknutý nosník, rovnoměrné zatížení, rozdílné hodnoty momentu I v levé a pravé polovině, vodorovná podložka níže než nosník).



Zadání:

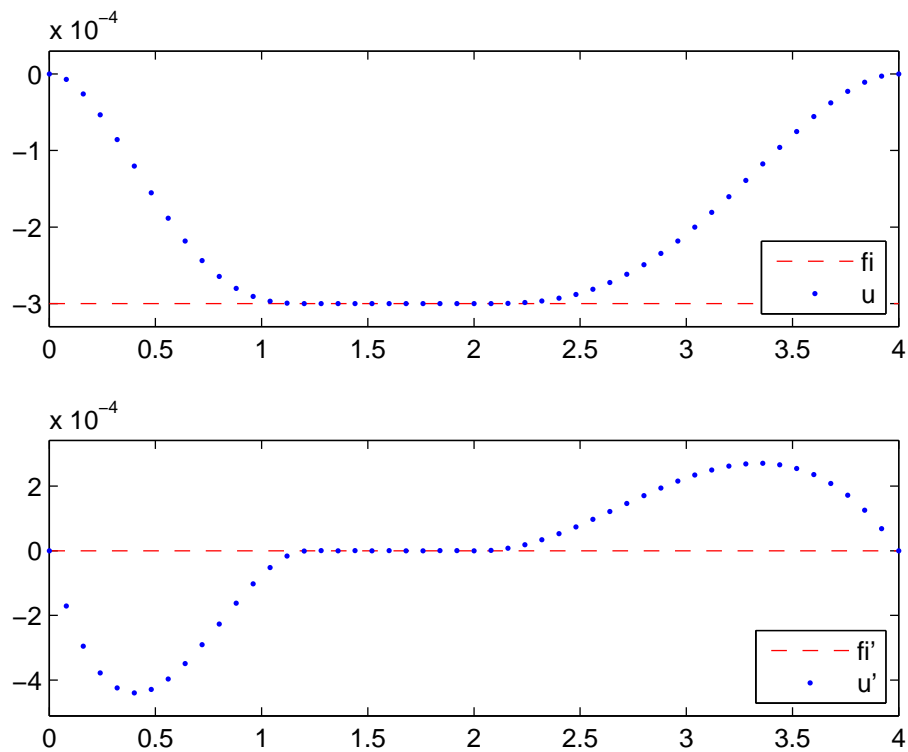
L	E	I	p	ε	φ
4	100	g	-1	10^{-7}	$-3 \cdot 10^{-4}$

Funkce g je ve tvaru

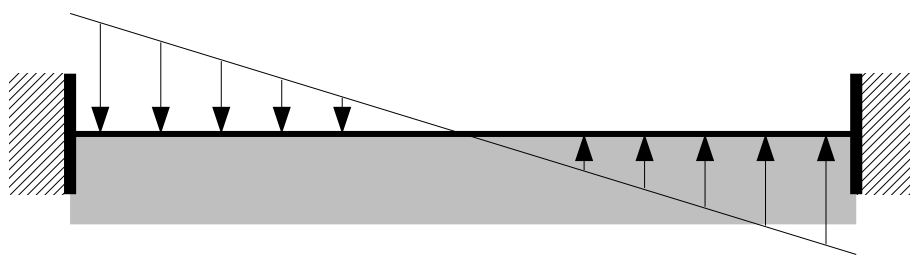
$$g(x) = \begin{cases} 1 & \text{pro } x \in [0, \frac{L}{2}], \\ 7 & \text{pro } x \in (\frac{L}{2}, L]. \end{cases}$$

Výsledky:

Počet prvků	30	50	80
Počet iterací	16	25	41
Body maximální výchylky	[0, 3; 0, 533333]	[0, 3; 0, 52]	[0, 3; 0, 525]
Hodnota maximální výchylky	$-3 \cdot 10^{-4}$	$-3 \cdot 10^{-4}$	$-3 \cdot 10^{-4}$

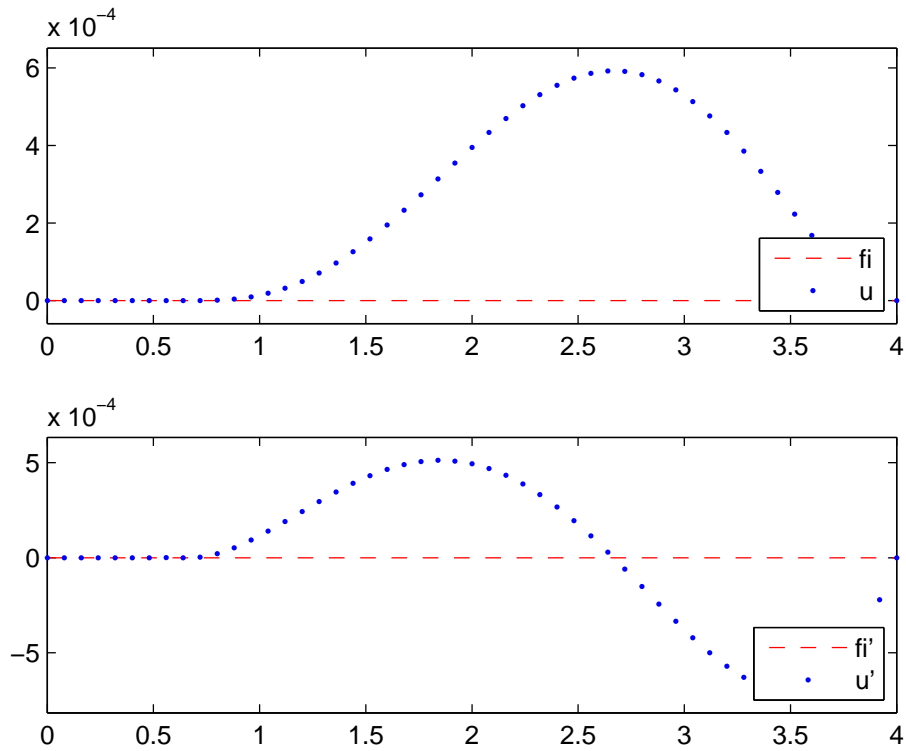


Příklad 6.6 (Oboustranně vetknutý nosník, lineární zatížení, vodorovná podložka).



Zadání:

L	E	I	p	ε	φ
4	100	1	$\frac{1}{2}x - 1$	10^{-7}	0



Výsledky:

Počet prvků	30	50	80
Počet iterací	11	17	28
Body maximální výchylky	0,666667	0,66	0,6625
Hodnota maximální výchylky	$5,9271 \cdot 10^{-4}$	$5,9203 \cdot 10^{-4}$	$5,9238 \cdot 10^{-4}$

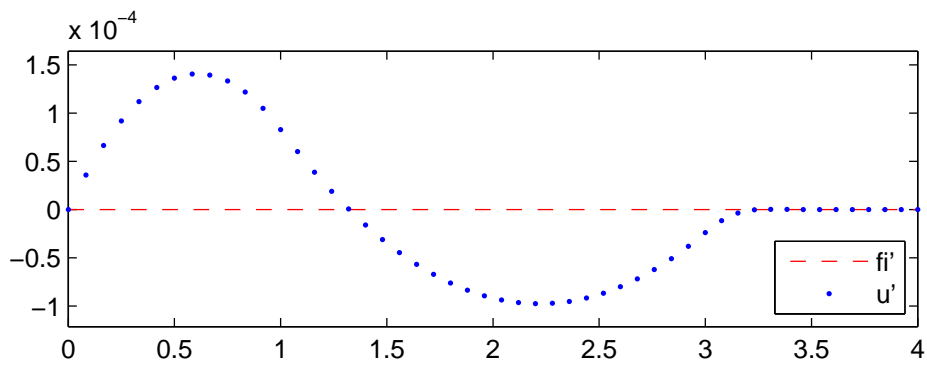
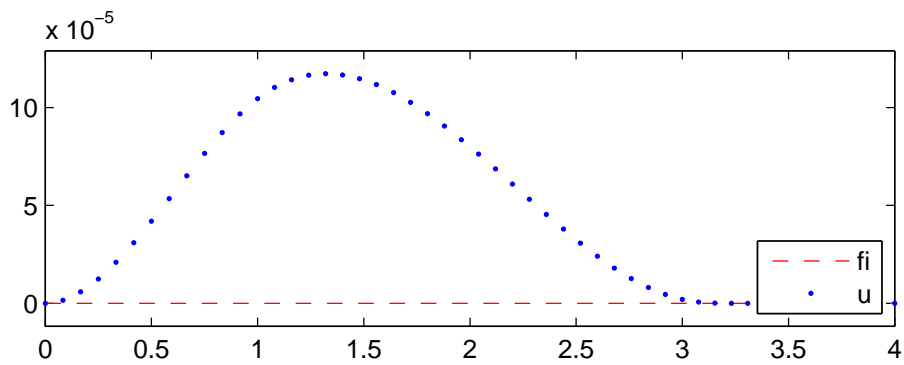
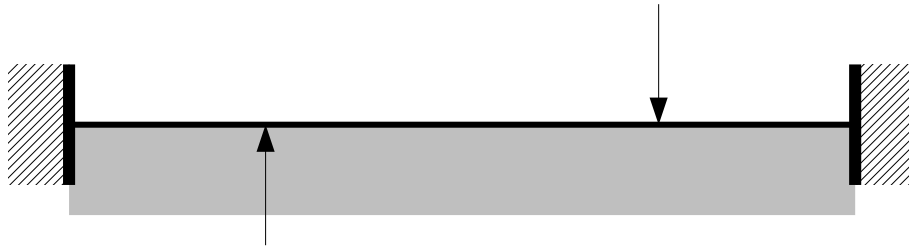
Příklad 6.7 (Oboustranně vetknutý nosník, dvě osamělé síly, vodorovná podložka).

Zadání:

L	E	I	p	ε	φ
4	100	1	0	10^{-7}	0

Vnější síly:

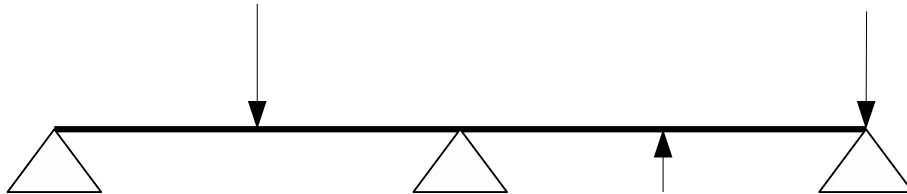
bod	$\frac{1}{4}L$	$\frac{3}{4}L$
hodnota	0,1	-0,1



Výsledky:

<i>Počet prvků</i>	30	50	80
<i>Počet iterací</i>	5	14	9
<i>Body maximální výchylky</i>	0,316667	0,33	0,335366
<i>Hodnota maximální výchylky</i>	$1,1687 \cdot 10^{-4}$	$1,1729 \cdot 10^{-4}$	$1,1719 \cdot 10^{-4}$

Příklad 6.8 (Na třech místech prostě podepřený nosník, tři osamělé síly).

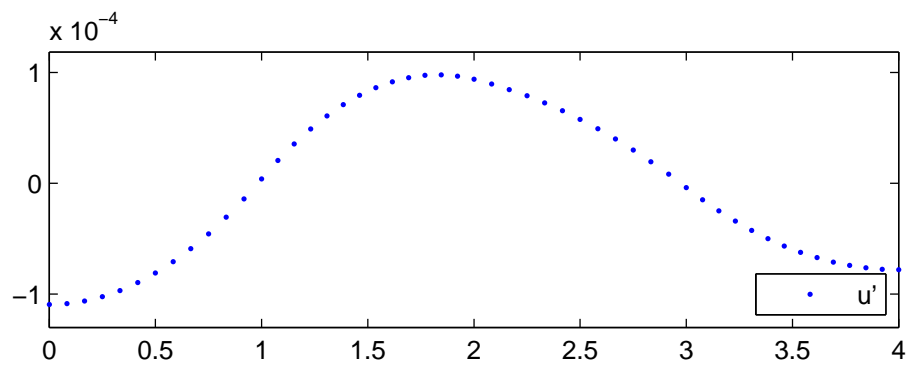
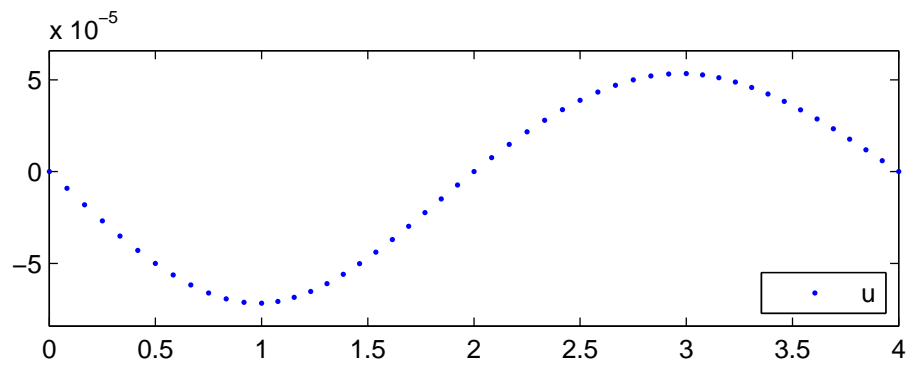


Zadání:

L	E	I	p	ε	φ
4	$2 \cdot 10^7$	1	0	10^{-6}	-

Vnější síly:

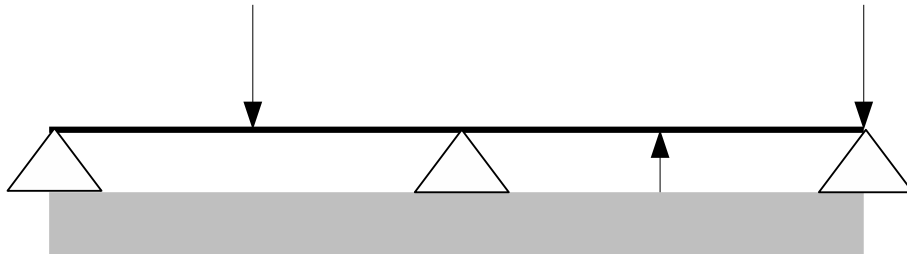
bod	$\frac{1}{4}L$	$\frac{3}{4}L$	1
hodnota	-10^4	$5 \cdot 10^3$	-10^4



Výsledky:

Počet prvků	30	50	80
Počet iterací	3	3	4
Body maximální výchylky	0,25	0,25	0,25
Hodnota maximální výchylky	$-7,1615 \cdot 10^{-5}$	$-7,1615 \cdot 10^{-5}$	$-7,1615 \cdot 10^{-5}$

Příklad 6.9 (Na třech místech prostě podepřený nosník, tři osamělé síly, vodorovná podložka níže než nosník).



Zadání:

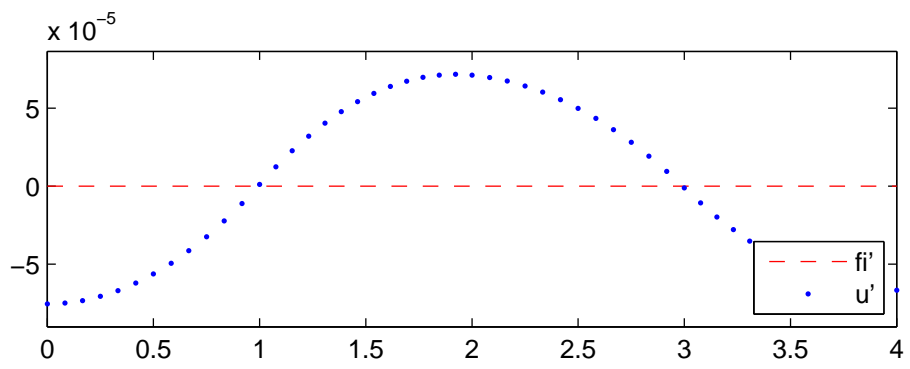
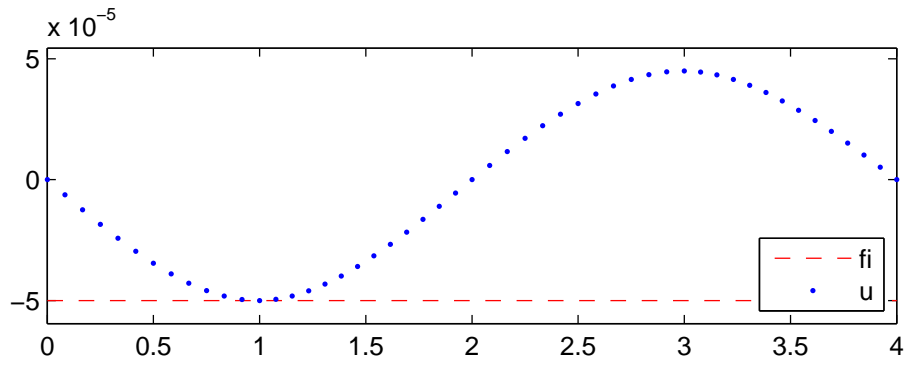
L	E	I	p	ε	φ
4	$2 \cdot 10^7$	1	0	10^{-7}	$-5 \cdot 10^{-5}$

Vnější síly:

bod	$\frac{1}{4}L$	$\frac{3}{4}L$	1
hodnota	-10^4	$5 \cdot 10^3$	-10^4

Výsledky:

Počet prvků	30	50	80
Počet iterací	1	1	1
Body maximální výchylky	0,25	0,25	0,25
Hodnota maximální výchylky	$-5 \cdot 10^{-5}$	$-5 \cdot 10^{-5}$	$-5 \cdot 10^{-5}$



Závěr

Nejdříve bych ráda zhodnotila výsledky z poslední kapitoly. Přes rozdílná zadání se počet iterací u příkladů 6.1 až 6.3 a 6.6 pohyboval kolem 12 a rostl se zvyšujícím se počtem prvků, až zhruba k 28 pro 80 prvků. V příkladě 6.5 bylo potřeba iterací o něco více, u 6.4 dokonce dvojnásobku, v příkladě 6.7 přibližně poloviny. Ač se v příkladě 6.8 nevyskytovala pevná překážka, bylo třeba 3, respektive 4 iterací. Je zajímavé ho srovnat s příkladem 6.9, který je vlastně tentýž, jen s překážkou, a byl dokončen během jediné iterace. Celý výpočet tedy odpovídal vyřešení lineární soustavy rovnic. Že překážka skutečně ovlivnila prohnutí nosníku, se lze přesvědčit srovnáním maximálních výchylek. Celkově z hlediska maximálních výchylek se jejich hodnoty a body, kde je jich nabýváno, měnily se změnou počtu prvků velmi málo, nebo vůbec.

Na začátku této práce byla stanovena rovnice a metody, které budou pro její řešení konstruovány. Byly představeny speciální nástroje pro zacházení s nehladkými funkcemi, použity na vytvoření algoritmů, a doplněny konvergenčními výsledky. Teorie byla aplikována na konkrétní úlohu, pro níž byl sestaven počítačový program, a s jeho pomocí byly provedeny numerické výpočty.

Domnívám se, že cíle práce byly splněny. Navíc tento text rozšířil česky psanou literaturu o nehladké Newtonově metodě, které je velmi poskrovnu. Jeho součástí tvoří variabilní počítačový program spolu s dokumentací zahrnující nápovědu a příklady zadání i s výsledky.

Mimo rámec práce, ale jejím logickým rozšířením a pokračováním, je zkoumání dalších speciálních tříd nehladkých funkcí, vytvoření odpovídajících algoritmů a jejich použití na konkrétních úlohách. Souvisejícím tématem jsou také globálně konvergentní newtonovské metody.

Při psaní této diplomové práce jsem získala mnoho cenných zkušeností, mezi něž patří skloubení teorie a numerických výpočtů, vytváření rozsáhlého programu, a snaha o jeho srozumitelnost a snadnou použitelnost.

Literatura

- [1] Clarke F. H.: *Optimization and Nonsmooth Analysis*, SIAM, Philadelphia, 1990.
- [2] Facchinei F., Pang J.-S.: *Finite-Dimensional Variational Inequalities And Complementarity Problems, Volume I and II*, Springer, New York, 2003.
- [3] Ferreira A. J. M.: *Matlab Codes For Finite Element Analysis*, Springer, 2009.
- [4] Glowinski, R.: *Numerical Methods For Nonlinear Variational Problems*, Springer, 2008.
- [5] Haslinger, J.: *Metoda konečných prvků pro řešení eliptických rovnic a nerovnic*, SPN, Praha, 1980.
- [6] Kwon Y. W., Bang H.: *The Finite Element Method Using MATLAB*, CRC Press, 1997.
- [7] Rachůnek L., Rachůnková I.: *Diferenciální počet funkcí více proměnných*, Univerzita Palackého v Olomouci, Olomouc, 2004.
- [8] Rektorys, K.: *Variační metody v inženýrských problémech a v problémech matematické fyziky*, Academia, Praha, 1999.
- [9] Ulbrich M.: *Nonsmooth Newton-like Methods for Variational Inequalities and Constrained Optimization Problems in Function Spaces*, Habilitation, TU München, Fakultät für Mathematik, 2002.