

Metody vnitřních bodů pro nekonvexní úlohy nelineárního programování

Ladislav Lukšan, Ctirad Matonoša, Jan Vlček
Ústav informatiky, AV ČR
e-mail: lukšan@cs.cas.cz

Nelineární programování

Obecná úloha nelineárního programování (NP) má tvar

$$F(x) \rightarrow \min, \quad c_I(x) \leq 0, \quad c_E(x) = 0,$$

kde $F : R^n \rightarrow R$, $c_I : R^n \rightarrow R^{m_I}$, $c_E : R^n \rightarrow R^{m_E}$ jsou dvakrát spojitě diferencovatelné funkce.

- Nelineární programování se používá k návrhu konstrukcí (jejich tvaru a topologie).
- Úlohy optimálního řízení vedou po diskretizaci na rozsáhlé úlohy nelineárního programování.

Metody vnitřních bodů

Metody vnitřních bodů byly původně vyvinuty pro řešení úloh lineárního programování

$$c^T x \rightarrow \min, \quad A^T x = b, \quad x \geq 0.$$

- Řešením je vrchol konvexního mnohostěnu.
- Simplexová metoda prohledává vrcholy konvexního mnohostěnu tak, aby docházelo ke snížení hodnoty účelové funkce.
- Počet vrcholů n -rozměrné krychle je 2^n .
- Klee a Minty (1972) ukázali příklad, kdy simplexová metoda projde všechny vrcholy "zkroucené" krychle \Rightarrow simplexová metoda je exponenciálně složitá.
- Khachian (1979) popsal polynomiálně složitou, avšak neefektivní elipsoidovou metodu pro lineární programování.
- Karmarkar (1984) navrhl polynomiálně složitou efektivní metodu vnitřních bodů pro lineární programování což podmínilo velký rozvoj metod vnitřních bodů od roku 1985.
- Gill a Murray (1986) ukázali, že Karmarkarova metoda je ekvivalentní Newtonově metodě používající logaritmickou barrierovou funkci, kterou zavedli Fiacco and Mc.Cormic (1968).
- Nemirovsky a Nesterov (1994) vyvinuli polynomiálně složitou metodu vnitřních bodů pro konvexní programování.
- Od roku 1995 se objevují metody vnitřních bodů pro řešení nekonvexních úloh nelineárního programování.

Nutné (KKT) podmínky pro (NP)

Nutné podmínky prvního řádu, kterým musí vyhovovat řešení úlohy (NP), mají tvar

$$\begin{aligned}g(x, u) &= 0, \\c_I(x) &\leq 0, \quad u_I \geq 0, \quad u_I^T c_I(x) = 0, \\c_E(x) &= 0,\end{aligned}$$

kde

$$g(x, u) = \nabla f(x) + A_I(x)u_I + A_E(x)u_E,$$

a $A_I(x) = [\nabla c_i(x) : i \in I]$, $A_E(x) = [\nabla c_i(x) : i \in E]$. Zde $u_I \in R^{m_I}$, $u_E \in R^{m_E}$ jsou vektory Lagrangeových multiplikátorů.

Princip metody vnitřních bodů (IP)

Řeší se posloupnost pomocných úloh

$$\begin{aligned}F(x) - \mu e^T \log(S_I)e &\rightarrow \min, \\c_I(x) + s_I &= 0, \\c_E(x) &= 0,\end{aligned}$$

kde $s_I > 0$ je vektor pomocných proměnných, e je vektor obsahující jednotkové prvky a $S_I = \text{diag}(s_i : i \in I)$ (předpokládáme, že $\mu \rightarrow 0$).

Nutné (KKT) podmínky pro (IP)

Nutné podmínky prvního řádu, kterým musí vyhovovat řešení úlohy (IP) jsou uvedeny v následující tabulce

primární formulace	primárně-duální formulace
$g(x, u) = 0,$	$g(x, u) = 0,$
$U_I e - \mu S_I^{-1} e = 0,$	$S_I U_I e - \mu e = 0,$
$c_I(x) + s_I = 0,$	$c_I(x) + s_I = 0,$
$c_E(x) = 0,$	$c_E(x) = 0,$

kde

$$g(x, u) = \nabla f(x) + A_I(x)u_I + A_E(x)u_E,$$

$S_I = \text{diag}(s_i : i \in I)$ a $U_I = \text{diag}(u_i : i \in I)$. Nerovnosti $s_i > 0$ a $u_i > 0$ musí být splněny ve všech iteracích. Primárně-duální formulace poskytuje efektivnější algoritmy.

Určení směrového vektoru

Linearizací KKT podmínek dostaneme krok Newtonovy metody:

$$\begin{bmatrix} G & 0 & A_I & A_E \\ 0 & U_I & S_I & 0 \\ A_I^T & I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta s_I \\ \Delta u_I \\ \Delta u_E \end{bmatrix} = - \begin{bmatrix} g \\ S_I U_I e - \mu e \\ c_I + s_I \\ c_E \end{bmatrix},$$

kde

$$g = g(x, u), \quad G = G(x, u) = \nabla^2 f(x) + \sum_{i \in I \cup E} u_i \nabla^2 c_i(x).$$

Předpokládáme, že tento systém je regulární.

Aktivní a neaktivní omezení

Soustava lineárních rovnic má výhodné vlastnosti, zavedeme-li aktivní a neaktivní omezení. Pro aktivní veličiny \hat{s}_I , \hat{u}_I a neaktivní veličiny \check{s}_I , \check{u}_I platí

$$\hat{s}_I \leq \varepsilon_I \hat{u}_I, \quad \check{s}_I > \varepsilon_I \check{u}_I,$$

kde $\varepsilon_I > 0$ je prahová hodnota. Eliminací pomocných proměnných a neaktivních omezení dostaneme výsledné rovnice

$$\begin{bmatrix} \hat{G} & \hat{A}_I & A_E \\ \hat{A}_I^T & -\hat{U}_I^{-1}\hat{S}_I & 0 \\ A_E^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \hat{u}_I \\ \Delta u_E \end{bmatrix} = - \begin{bmatrix} \hat{g} \\ \hat{c}_I + \mu \hat{U}_I^{-1}e \\ c_E \end{bmatrix},$$

kde

$$\hat{g} = g + \check{A}_I \check{S}_I^{-1} \check{U}_I \check{c}_I + \mu \check{A}_I \check{S}_I^{-1} e, \quad \hat{G} = G + \check{A}_I \check{S}_I^{-1} \check{U}_I \check{A}_I^T.$$

Indefinitní předpodmiňovač

Výsledný lineární KKT systém lze zapsat ve tvaru

$$K \bar{d} = \begin{bmatrix} \hat{G} & \hat{A} \\ \hat{A}^T & -\hat{M} \end{bmatrix} \begin{bmatrix} d \\ \hat{d} \end{bmatrix} = \begin{bmatrix} b \\ \hat{b} \end{bmatrix} = \bar{b},$$

kde \hat{M} je pozitivně semidefinitní diagonální matice. Předpokládáme, že matice K je regulární. Tento lineární KKT systém lze s výhodou předpominít maticí

$$C = \begin{bmatrix} \hat{D} & \hat{A} \\ \hat{A}^T & -\hat{M} \end{bmatrix},$$

kde \hat{D} je pozitivně definitní diagonální matice (odvozená obvykle z diagonály matice \hat{G}). Tato matice má podobnou strukturu jako matice K . Je však řidší a při jejím rozkladu nedochází k tak velkému zaplnění jako v případě matice K .

Výhodu tohoto předpodmiňovače demonstrují následující tři věty:

Věta 1. Matice KC^{-1} má alespoň $\hat{n}_I + 2m_E$ jednotkových vlastních čísel, ale nanejvýš $\hat{n}_I + m_E$ jim odpovídajících lineárně nezávislých vlastních vektorů. Zbylá vlastní čísla matice KC^{-1} jsou vlastními čísly matice

$$Z_E^T \tilde{G} Z_E (Z_E^T \tilde{D} \tilde{Z}_E)^{-1},$$

kde Z_E je matice jejíž sloupce tvoří bázi ortogonálního doplňku podprostoru generovaného sloupci matice A_E a

$$\tilde{G} = \hat{G} + \hat{A}_I \hat{M}_I^{-1} \hat{A}_I^T, \quad \tilde{D} = \hat{D} + \hat{A}_I \hat{M}_I^{-1} \hat{A}_I^T.$$

Věta 2. Krylovův podprostor \mathcal{K} definovaný maticí KC^{-1} má dimenzi nejvýše $\min(n+1, n-m_E+2)$.

Věta 3. Uvažujme algoritmus PCG s předpodmiňovačem C aplikovaný na systém $K\bar{d} = \bar{b}$. Nechť počáteční vektor \bar{d} je zvolen tak, že $\hat{r} = 0$ a nechť matice $Z_E^T \tilde{G} Z_E$ je pozitivně definitní. Pak:

- Vektor d^* (první část vektoru \bar{d}^* , který je řešením rovnice $K\bar{d} = \bar{b}$) je nalezen po nejvýše $n - m_E$ iteracích.
- Algoritmus nemůže selhat, dříve než je nalezen vektor d^* .

- Chyba $\|d - d^*\|$ konverguje k nule přinejmenším R - lineárně s kvocientem

$$\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1},$$

kde κ je spektrální číslo podmíněnosti matice $Z_E^T \tilde{G} Z_E (Z_E^T \tilde{D} Z_E)^{-1}$.

- Jestliže $d = d^*$, pak také $\hat{d}_I = \hat{d}_I^*$ a vektor d_E^* můžeme určit vztahem

$$d_E^* = d_E + (A_E^T \tilde{D}^{-1} A_E)^{-1} A_E^T \tilde{D}^{-1} r.$$

Pokutová funkce pro výběr délek kroku

Při výběru délky kroku je třeba použít vhodnou pokutovou funkci, která klesá ve směru získaném řešením lineárního KKT systému. Výhodné vlastnosti má rozšířená Lagrangeova funkce

$$\begin{aligned} P(\alpha) &= f(x + \alpha \Delta x) - \mu e^T \ln(S_I(\alpha))e \\ &+ (u_I + \Delta u_I)^T (c_I(x + \alpha \Delta x) + s_I(\alpha)) \\ &+ (u_E + \Delta u_E)^T c_E(x + \alpha \Delta x) \\ &+ \frac{\sigma}{2} \|c_I(x + \alpha \Delta x) + s_I(\alpha)\|^2 \\ &+ \frac{\sigma}{2} \|c_E(x + \alpha \Delta x)\|^2, \end{aligned}$$

kde $\sigma \geq 0$ je pokutový parametr. Hodnota tohoto parametru by měla být co nejnižší.

Věta 4. Nechť $s_I > 0$, $u_I > 0$ a nechť vektory Δx , Δs_I jsou nepřesným řešením lineárního KKT systému. Pak

$$\begin{aligned} P'(0) &= -(\Delta x)^T G \Delta x - (\Delta s_I)^T S_I^{-1} U_I \Delta s_I \\ &- \sigma (\|c_I + s_I\|^2 + \|c_E\|^2) \\ &+ (\Delta x)^T r + \sigma ((\hat{c}_I + \hat{s}_I)^T \hat{r}_I + c_E^T r_E). \end{aligned}$$

kde r , \hat{r}_I , r_E jsou části residuálního vektoru. Jestliže

$$\sigma > -\frac{(\Delta x^T) G \Delta x + (\Delta s_I)^T S_I^{-1} U_I \Delta s_I}{\|c_I + s_I\|^2 + \|c_E\|^2}$$

a člen $(\Delta x)^T r + \sigma ((\hat{c}_I + \hat{s}_I)^T \hat{r}_I + c_E^T r_E)$ je dostatečně malý (přesnost řešení), pak

$$P'(0) < 0.$$

Restart

Pokud $P'(0) \geq 0$, nelze nalézt vhodnou délku kroku. Máme dvě možnosti.

- Změníme hodnotu parametru $\sigma \geq 0$ tak, aby byla splněna nerovnost

$$\sigma > -\frac{(\Delta x^T) G \Delta x + (\Delta s_I)^T S_I^{-1} U_I \Delta s_I}{\|c_I + s_I\|^2 + \|c_E\|^2}.$$

Pak $P'(0) < 0$, pokud řešíme lineární KKT systém s dostatečnou přesností.

- Ponecháme $\sigma \geq 0$ beze změny, nahradíme matici \hat{G} pozitivně definitní diagonální maticí \hat{D} a řešíme nový systém. Obvykle použijeme stejnou diagonální matici jako při konstrukci indefinitního předpodmiňovače. Použití restartů je mnohem efektivnější než přepočítávání parametru $\sigma \geq 0$.

Určení barierového parametru

Hodnota barierového parametru μ se obvykle určuje tak, aby platilo $0 < \mu < s_I^T u_I / m_I$ (neboli $\mu = \lambda s_I^T u_I / m_I$, kde $0 < \lambda < 1$).

Výpočetní zkušenosti naznačují, že algoritmus funguje nejlépe, když veličiny $s_i u_i$ konvergují k nule pokud možno stejně rychle. Odchylku od této uniformity lze měřit podílem

$$\varrho = \frac{\min_{i \in I} (s_i u_i)}{s_I^T u_I / m_I}$$

(míra centrality). Zřejmě $0 < \varrho \leq 1$ a $\varrho = 1$, právě když $S_I U_I e = \mu e$. Hodnota λ (a tedy μ) se pak určuje pomocí ϱ . Používají se obvykle různé heuristické vzorce.

Numerické experimenty

Metody vnitřních bodů byly testovány pomocí 17 testovacích problémů s 1000 proměnnými (Test 18). Tento test je uveden na stránce

www.cs.cas.cz/~luksan/test.html

Testované realizace

- M - metoda pro výběr délky kroku (N - použití počáteční délky kroku, P - použití pokutové funkce, B - použití bariérového filtru s 50 vzorky, M - použití Markovova filtru s jedním vzorkem).
- S - první (1) nebo druhá (2) strategie pro výběr délky kroku.
- P - indefinitní předpodmiňovač s úplným (1) nebo neúplným (2) Gillovým–Murrayovým rozkladem.

Význam hodnot uvedených v tabulkách

- NIT - počet iterací.
- NFV - počet použitých funkčních hodnot.
- NFG - počet použitých gradientů.
- NCG - počet vnitřních iterací metody CG.
- NRS - počet restartů.
- TIME - celkový čas výpočtu (pro 18 testovacích problémů).
- NFAIL - počet selhání (počet nevyřešených problémů).

Výsledky testů

Testování pomocí testovacích problémů systému UFO:

M	S	P	NIT	NFV	NFG	NCG	NRS	TIME	NFAIL
N	1	1	534	535	3735	2795	9	3.11	1
N	1	2	535	537	3748	3147	7	2.92	1
N	2	1	567	567	4114	2993	6	3.09	-
N	2	2	582	582	4219	3006	7	2.91	-
P	1	1	511	629	3902	4664	38	3.59	1
P	1	2	481	638	3602	4886	27	3.22	1
P	2	1	540	637	4070	3475	21	3.52	-
P	2	2	575	707	4408	3738	30	3.53	-
B	2	1	513	565	3779	2653	25	3.27	-
B	2	2	527	596	3877	2756	21	3.05	-
M	2	1	504	590	3700	2525	10	3.35	-
M	2	2	536	632	3924	2797	13	3.17	-

Testování pomocí prostředí CUTE:

Problem	n	m	S	P	NIT	NFV	NFG	NCG
BRITGAS	450	360	1	1	15	15	285	132
CLNLBEAM	1503	1000	1	1	19	19	133	81
DALLASL	906	667	1	1	47	47	893	47
EG3	1001	2000	2	2	41	41	287	251
EIGENC2	462	231	1	1	17	17	7531	180
GAUSSELM	819	1296	2	1	20	20	660	1640
HANGING	1800	1150	1	1	29	29	609	792
MANNE	600	400	2	2	50	50	300	476
NGONE	100	1273	2	2	35	35	3535	539
READING1	2002	1000	2	2	35	35	245	352
READING3	2002	1001	2	2	19	19	133	532
READING4	1001	1000	2	3	51	51	204	73
SINROSNB	1000	999	1	1	13	13	52	50
SREADIN3	1002	501	1	2	38	38	266	193
SSNLBEAM	3003	2000	1	1	19	19	133	125
SVANBERG	1000	1000	1	1	20	20	380	81
TRAINF	2008	1002	1	1	37	37	370	94
TRAINH	2008	1002	1	1	30	30	390	424
ZAMB2	1326	480	1	2	29	29	348	1927

Srovnání s programem KNITRO:

Problem	Algorithm 1			KNITRO		
	n	m	NFV	n	m	NFV
CLNLBEAM	1503	1000	19	303	200	21
DALLASL	906	667	47	906	667	100
EG3	1001	2000	41	101	200	31
GAUSSELM	819	1926	22	819	1926	115
GRIDNETA	924	484	12	924	484	21
GRIDNETI	924	484	15	924	484	28
MANNE	600	400	50	300	200	9
NGONE	100	1273	35	100	1273	217
READING1	2002	1000	35	202	100	52
READING3	2002	1001	19	303	200	12
READING4	1001	1000	51	202	101	77
SINROSNB	1000	999	13	1000	999	90
SREADIN3	1002	501	38	202	101	30
SSNLBEAM	3003	2000	19	303	200	23
SVANBERG	1000	1000	20	1000	1000	18
TRAINF	2008	1002	34	808	402	345
TRAINH	2008	1002	30	808	402	441
ZAMB2	1326	480	29	1326	480	37

Publikace

- Lukšan L., Vlček J.: Indefinitely Preconditioned Inexact Newton Method for Large Sparse Equality Constrained Nonlinear Programming Problems. *Numerical Linear Algebra with Applications* 5 (1998) 219-247.
- Lukšan L., Matonoha C., Vlček J.: Interior point method for nonlinear nonconvex optimization. *Numerical Linear Algebra with Applications* 11 (2004) 431-453.
- Lukšan L., Matonoha C., Vlček J.: Interior point methods for large-scale nonlinear programming. *Optimization Methods and Software* 20 (2005) 569-582.