národní
úložiště
šedé
literatury

Optimalizace osvitu pro tepelný ohřev forem v automobilovém průmyslu

Královcová, J.
2009

Dostupný z http://www.nusl.cz/ntk/nusl-40026

Dílo je chráněno podle autorského zákona č. 121/2000 Sb.

Tento dokument byl stažen z Národního úložiště šedé literatury (NUŠL).

Datum stažení: 27.04.2024

Další dokumenty můžete najít prostřednictvím vyhledávacího rozhraní nusl.cz .

Institute of Computer Science
Academy of Sciences of the Czech Republic

Optimalizace osvitu pro tepelný ohřev forem v automobilovém průmyslu

J.Královcová, L.Lukšan, J.Mlýnek

Technical report No. V1050

Únor 2009

Pod Vodárenskou věží 2, 182 07 Prague 8 phone: +420 2 688 42 44, fax: +420 2 858 57 89,
e-mail:ics@cs.cas.cz

Institute of Computer Science
Academy of Sciences of the Czech Republic

Optimalizace osvitu pro tepelný ohřev forem v automobilovém průmyslu

J.Královcová, L.Lukšan, J.Mlýnek [1]

Technical report No. V1050

Únor 2009

Abstrakt:

V práci je navržen model osvitu hliníkových forem používaných v automobilovém průmyslu. Tento model je analyzován a jsou odvozeny jeho citlivosti na změnu parametrů zahřívacích lamp, které slouží k výpočtu gradientu účelové funkce. Účelová funkce má tvar součtu čtverců a je třeba ji minimalizovat na množině zadané nelineárními omezeními ve tvaru rovností. V práci je popsána optimalizační metoda sloužící k optimálnímu nastavení parametrů zahřívacích lamp, jsou uvedeny procedury realizující účelovou funkci a omezující funkce a jsou ukázány výsledky získané aplikací tohoto postupu v případě modelových úloh.

Keywords:
model osvitu plochy, optimalizace, nelineární programování, metoda rekursivního kvadratického programování, softwarová realizace, optimalizační systém UFO.

[1] Tato práce byla vytvořena v rámci centra excelence MŠMT 1M0554 a podpořena výzkumným záměrem AV0Z10300504 Akademie věd ČR.

1 Úloha rovnoměrného zahřívání formy

Zadání úlohy:

- Uvažujeme hliníkovou formu váhy cca 300 kg. Požadujeme rovnoměrné zahřívání formy pomocí 100 lamp stejného výkonu na teplotu 270°C.
- Každá lampa je zadána pomocí kartézských souřadnic krajních bodů A, B a souřadnic vektoru směru svitu lampy u, délka d všech lamp je stejná.
- Plocha formy je zadána pomocí plošných elementů. Každý element je reprezentován souřadnicemi těžiště T a souřadnicemi vnější normály v k plošnému elementu v bodě T.
- Je stanovena výchozí poloha každé lampy (ta je zadána pomocí 9 parametrů). Cílem je zajistit rovnoměrné zahřívání formy na základě vhodného nastavení uvedených parametrů lamp.
- Souřadnice každého těžiště T příslušného elementu a příslušné vnější normály v jsou pevně dány pomocí 6 parametrů.

Předpokládaná forma zadání parametrů lampy ve vstupním souboru:

$x_A, y_A, z_A, x_B, y_B, z_B, x_u, y_u, z_u$ – první tři hodnoty udávají kartézské souřadnice krajního bodu A lampy, další tři hodnoty udávají kartézské souřadnice krajního bodu B lampy a poslední tři hodnoty udávají souřadnice vektoru směru svitu lampy.

Předpokládaná forma zadání parametrů plošného elementu ve vstupním souboru:

$x_T, y_T, z_T, x_v, y_v, z_v$ – první tři hodnoty udávají kartézské souřadnice těžiště T plošného elementu, další tři hodnoty udávají souřadnice vnější normály k ploše v bodě T.

Značení:

- Bod C_k reprezentuje „svítící bod“ lampy (viz Obrázek 1.1).
- Označme α_k velikost úhlu, který svírá přímka určená body C_k, T se směrem svitu lampy u, dále označme β_k velikost úhlu, který svírá přímka určená body C_k, T s vnější normálou v (viz Obrázek 1.1).

Omezující předpoklad:

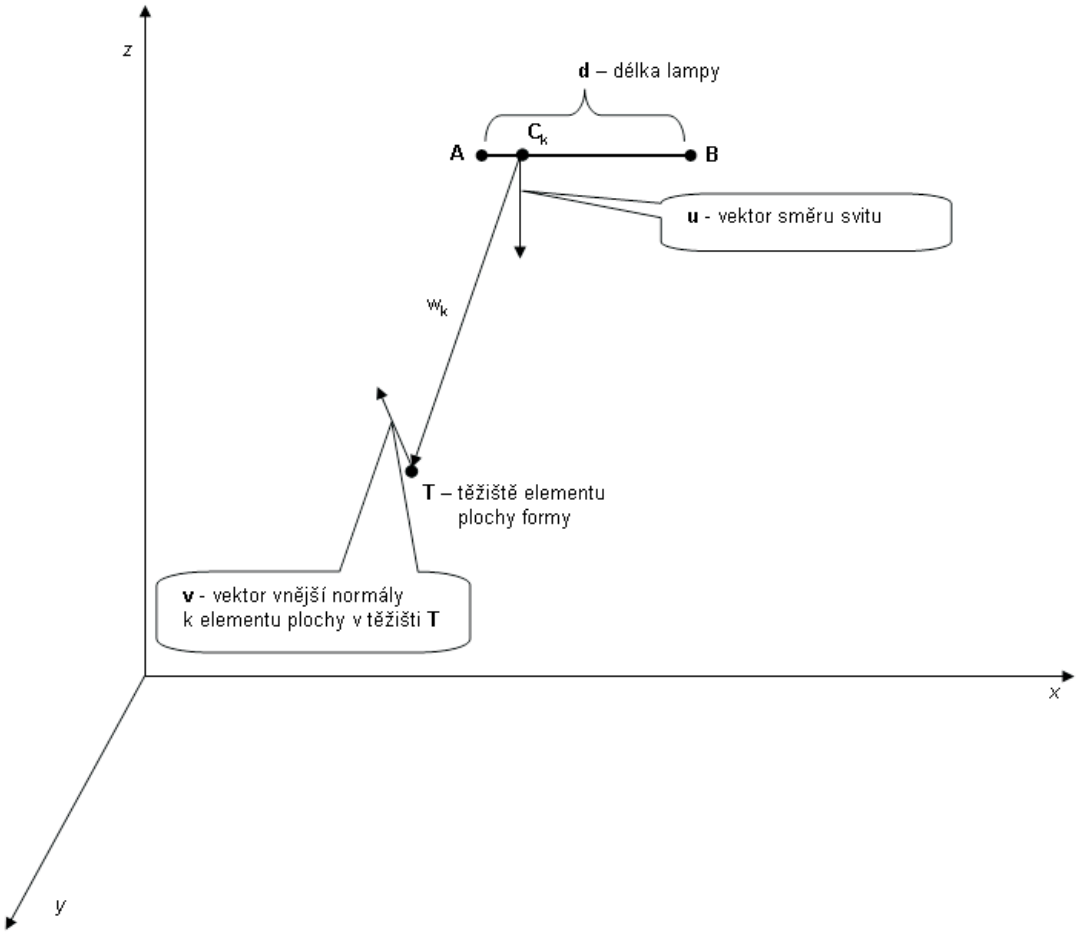
Pokud pro uvažovaný „svítící bod“ C_k a bod T platí, že úhel $\alpha > \pi/4$ nebo úhel $\beta > \pi/4$, pak předpokládáme, že plocha zadaná těžištěm T není adekvátní částí lampy kolem bodu C_k zahřívána.

Distribuční funkce intenzity svitu:

Uvedená funkce je uvažována ve tvaru $f(\alpha) = a\cos(\alpha) + b\,|\sin(\alpha)|$, kde $a, b \in R$ a $0 \leq \alpha \leq \pi/4$. Pro další uvedený postup je užita volba $a = 3$ a $b = 0.5$, tj.

$$f(\alpha) = 3\cos(\alpha) + 0.5\,|\sin(\alpha)|\,,$$

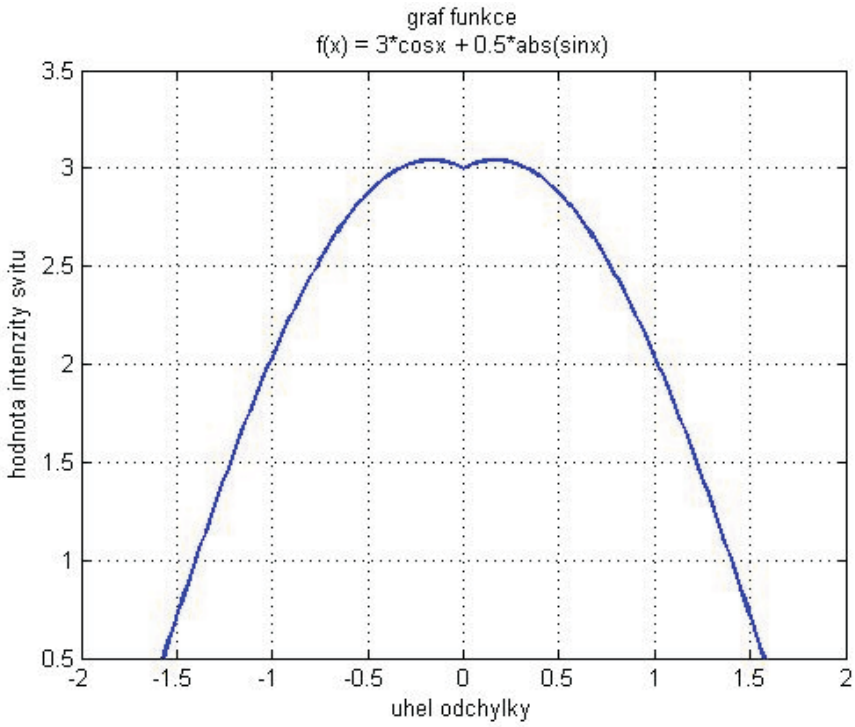
viz Obrázek 1.2. Tato funkce nabývá lokálního minima pro $\alpha = 0$ a maxima pro $\alpha \cong 0.165$.



Obrázek 1.1:

Výpočet intenzity svitu lampy AB na plochu určenou těžištěm elementu T:

- Označme p počet uvažovaných „svítících bodů“ C_k, $k = 0, 1, \ldots, p-1$, dané lampy (p je liché číslo), kde $C_0 = A$ a $C_{p-1} = B$.
- Délka vektoru směru svitu je $\|u\| = \sqrt{x_u^2 + y_u^2 + z_u^2}$.
- Délka vektoru vnější normály v bodě T je $\|v\| = \sqrt{x_v^2 + y_v^2 + z_v^2}$.



Obrázek 1.2:

- Výpočet vzdálenosti mezi dvěma body C_k, C_{k-1}:

$$h = \sqrt{\left(\frac{x_B - x_A}{p-1}\right)^2 + \left(\frac{y_B - y_A}{p-1}\right)^2 + \left(\frac{z_B - z_A}{p-1}\right)^2}.$$

- Pro $k = 0, 1, \ldots, p-1$ provedeme následující výpočet:

  1) Určení souřadnic bodu C_k:

$$x_{C_k} = x_A + \frac{x_B - x_A}{p-1} \cdot k\,,$$
$$y_{C_k} = y_A + \frac{y_B - y_A}{p-1} \cdot k\,,$$
$$z_{C_k} = z_A + \frac{z_B - z_A}{p-1} \cdot k\,.$$

  2) Určení souřadnic vektoru svitu w_k bodu C_k do těžiště T:

$$x_{w_k} = x_T - x_{C_k}\,,$$
$$y_{w_k} = y_T - y_{C_k}\,,$$
$$z_{w_k} = z_T - z_{C_k}\,.$$

Výpočet délky vektoru w_k:

$$\|w_k\| = \sqrt{x_{w_k}^2 + y_{w_k}^2 + z_{w_k}^2}.$$

3) Odchylka α_k vektorů u a w_k:

$$\cos\alpha_k = \frac{u \cdot w_k}{\|u\| \cdot \|w_k\|}.$$

4) Odchylka β_k vektorů v a w_k:

$$\cos\beta_k = \frac{(-v) \cdot w_k}{\|v\| \cdot \|w_k\|}.$$

5) Intenzita záření I_k ve směru vektoru w_k z bodu C_k do T užitím distribuční funkce intenzity záření dané vztahem $f(\alpha) = 3\cos(\alpha) + 0.5\,|\sin(\alpha)|$:

$$I_k = \frac{3\cos\alpha_k + 0.5\sqrt{1-\cos^2\alpha_k}}{\|w_k\|^2} \cdot d_k \cdot \cos\beta_k,$$

kde $d_k = h/2$ pro $k = 0, p-1$; $d_k = h$ pro $k = 1, \ldots, p-2$.

6) Pokud $\cos\alpha_k < \frac{\sqrt{2}}{2}$ nebo $\cos\beta_k < \frac{\sqrt{2}}{2}$, pokládáme $I_k = 0$.

- Celková intenzita I svitu lampy AB na plochu s těžištěm T:

$$I = \sum_{k=0}^{p-1} I_k = \sum_{k=0}^{p-1} \frac{3\cos\alpha_k + 0.5\sqrt{1-\cos^2\alpha_k}}{\|w_k\|^2} \cdot \cos\beta_k \cdot d_k =$$
$$= \sum_{k=0}^{p-1} \frac{\frac{3u\cdot w_k}{\|u\|\cdot\|w_k\|} + 0.5\sqrt{1-(\frac{u\cdot w_k}{\|u\|\cdot\|w_k\|})^2}}{\|w_k\|^2} \cdot \frac{(-v)\cdot w_k}{\|v\|\cdot\|w_k\|} \cdot d_k$$

Hodnota I závisí na 9 volných parametrech (souřadnice krajních bodů A, B lampy a souřadnice vektoru u směru svitu lampy) a na 6 parametrech pevných (souřadnice těžiště elementu plochy T a souřadnice vektoru vnější normály v).

Součástí zadání praktické úlohy bude i „strukturovaná matice osvícení“. Pro výchozí pozice umístění lamp a jejich směru svícení bude zadána matice, kde každému prvku matice odpovídajícímu dané lampě a elementu plochy bude přiřazena hodnota 1, pokud výchozí intenzita svitu lampy na element plochy je nenulová, v opačném případě bude prvku přiřazena hodnota 0.

2 Formulace optimalizační úlohy

Rovnice pro osvit elementu formy lampou

Nejprve budeme uvažovat závislost osvitu jednoho elementu formy jednou lampou. Těžiště elementu formy má souřadnice $x^T = (x_1^T, x_2^T, x_3^T)$ a jeho normálový vektor má souřadnice $x^N = (x_1^N, x_2^N, x_3^N)$. V dalších úvahách budeme používat označení $v = -x^N$. Krajní body lampy mají souřadnice $x^A = (x_1^A, x_2^A, x_3^A)$, $x^B = (x_1^B, x_2^B, x_3^B)$. Vektor směru svitu lampy má souřadnice $x^S = (x_1^S, x_2^S, x_3^S)$. V dalších úvahách budeme používat označení $u = x^S$. Aby byly dodrženy rozměry lampy a aby nedocházelo k nejednoznačnostem, budeme uvažovat omezení

$$\sum_{i=1}^{3} (x_i^S)^2 = 1,$$

$$\sum_{i=1}^{3} x_i^S (x_i^B - x_i^A) = 0,$$

$$\sum_{i=1}^{3} (x_i^B - x_i^A)^2 = d^2,$$

kde d je délka lampy. První omezení určuje jednotkovou délku vektoru směru svitu lampy, druhé jeho kolmost k ose lampy a třetí délku lampy.

Lampu lze modelovat jako lineární útvar délky d, skládající se z p svítících elementů, které mají délky d_k, $1 \le k \le p$. Přitom $d_k = d/(p-1)$, $1 < k < p$, a délky krajních elementů jsou poloviční. Vzdálenosti středů svítících elementů od těžiště elementu formy lze vyjádřit vzorcem

$$w_k = x^T - (1 - \lambda_k) x^A - \lambda_k x^B, \quad \lambda_k = \frac{k-1}{p-1},$$

kde $1 \le k \le p$. Použijeme-li zavedené označení, můžeme modelové rovnice uvedené v první části práce zapsat ve tvaru

$$I = \sum_{k=1}^{p} I_k,$$

kde

$$I_k = \left(3\alpha_k + \frac{1}{2}\sqrt{1 - \alpha_k^2}\right) \frac{\beta_k}{\|w_k\|^2} d_k,$$

$$\alpha_k = \frac{u^T w_k}{\|u\| \|w_k\|} = \tilde{u}^T \tilde{w}_k,$$

$$\beta_k = \frac{v^T w_k}{\|v\| \|w_k\|} = \tilde{v}^T \tilde{w}_k,$$

a kde $\tilde{u} = u/\|u\|$, $\tilde{v} = v/\|v\|$, $\tilde{w}_k = w_k/\|w_k\|$ jsou normalizované vektory (v této části práce budeme používat označení α_k a β_k místo $\cos\alpha_k$ a $\cos\beta_k$).

Abychom mohli použít efektivní optimalizační metodu, je třeba znát analytické vyjádření derivací osvitu I podle složek vektorů x^A, x^B, x^S (složky vektorů x^T, x^N jsou konstantní neboť forma se nemění). Platí

$$\begin{aligned}
\frac{\partial I}{\partial x_i^A} &= \sum_{k=1}^{p} \frac{\partial I_k}{\partial x_i^A} = -\sum_{k=1}^{p} (1-\lambda_k) \frac{\partial I_k}{\partial w_{ik}}, \\
\frac{\partial I}{\partial x_i^B} &= \sum_{k=1}^{p} \frac{\partial I_k}{\partial x_i^B} = -\sum_{k=1}^{p} \lambda_k \frac{\partial I_k}{\partial w_{ik}} \\
\frac{\partial I}{\partial x_i^S} &= \sum_{k=1}^{p} \frac{\partial I_k}{\partial x_i^S} = \sum_{k=1}^{p} \frac{\partial I_k}{\partial u_i},
\end{aligned}$$

kde

$$\begin{aligned}
\frac{\partial I_k}{\partial u_i} &= \left(3 - \frac{1}{2}\frac{\alpha_k}{\sqrt{1-\alpha_k^2}}\right) \frac{\beta_k}{\|w_k\|^2} \frac{\partial \alpha_k}{\partial u_i} d_k, \\
\frac{\partial I_k}{\partial w_{ik}} &= \left(3 - \frac{1}{2}\frac{\alpha_k}{\sqrt{1-\alpha_k^2}}\right) \frac{\beta_k}{\|w_k\|^2} \frac{\partial \alpha_k}{\partial w_{ik}} d_k \\
&+ \left(3\alpha_k + \frac{1}{2}\sqrt{1-\alpha_k^2}\right) \left(\frac{1}{\|w_k\|^2}\frac{\partial \beta_k}{\partial w_{ik}} - 2\frac{\beta_k}{\|w_k\|^4} w_{ik}\right) d_k.
\end{aligned}$$

Dále platí

$$\begin{aligned}
\frac{\partial \alpha_k}{\partial u_i} &= \frac{w_{ik}}{\|u\|\|w_k\|} - \frac{u^T w_k}{\|u\|\|w_k\|}\frac{u_i}{\|u\|^2} = \frac{1}{\|u\|}(\tilde{w}_{ik} - \alpha_k \tilde{u}_i), \\
\frac{\partial \alpha_k}{\partial w_{ik}} &= \frac{u_i}{\|u\|\|w_k\|} - \frac{u^T w_k}{\|u\|\|w_k\|}\frac{w_{ik}}{\|w_k\|^2} = \frac{1}{\|w_k\|}(\tilde{u}_i - \alpha_k \tilde{w}_{ik}), \\
\frac{\partial \beta_k}{\partial w_{ik}} &= \frac{v_i}{\|v\|\|w_k\|} - \frac{v^T w_k}{\|v\|\|w_k\|}\frac{w_{ik}}{\|w_k\|^2} = \frac{1}{\|w_k\|}(\tilde{v}_i - \beta_k \tilde{w}_{ki}),
\end{aligned}$$

a po dosazení dostaneme

$$\begin{aligned}
\frac{\partial I_k}{\partial u_i} &= \left(3 - \frac{1}{2}\frac{\alpha_k}{\sqrt{1-\alpha_k^2}}\right) \frac{\beta_k d_k}{\|u\|\|w_k\|^2}(\tilde{w}_{ik} - \alpha_k \tilde{u}_i) \\
\frac{\partial I_k}{\partial w_{ik}} &= \left(3 - \frac{1}{2}\frac{\alpha_k}{\sqrt{1-\alpha_k^2}}\right) \frac{\beta_k d_k}{\|w_k\|^3}(\tilde{u}_i - \alpha_k \tilde{w}_{ik}) \\
&+ \left(3\alpha_k + \frac{1}{2}\sqrt{1-\alpha_k^2}\right) \frac{d_k}{\|w_k\|^3}(\tilde{v}_i - 3\beta_k \tilde{w}_{ik}).
\end{aligned}$$

Toto jsou všechny vztahy, které potřebujeme k určení hodnoty a gradientu účelové funkce popsané v následujícím oddílu. Všimněne si, že není třeba znát složky vektorů u, v a w_k, $1 \le k \le p$. Potřebujeme pouze jejich eukleidovské normy a složky normalizovaných vektorů $\tilde{u}$, $\tilde{v}$ a $\tilde{w}_k$, $1 \le k \le p$. Proto je účelné vektory u, v a w_k, $1 \le k \le p$, předem normalizovat. Výpočet hodnoty I a jejích derivací realizuje procedura COMPL. uvedená v poslední části práce.

Účelová funkce a omezení zajišťující rovnoměrný osvit formy

Předpokládejme, že máme n_e elementů formy a n_l lamp. Každý element může být osvícen více lampami. Nechť L_j je množina indexů lamp svítících na j-tý element. Zvolme $1 \leq j \leq n_e$ a $l \in L_j$. Označíme-li I_{jl} intenzitu osvitu j-tého prvku l-tou lampou (tato hodnota odpovídá hodnotě I z předchozího oddílu), pak celková intenzita osvitu j-tého prvku I_j je dána vzorcem

$$I_j = \sum_{l \in L_j} I_{jl}$$

a její derivace můžeme vypočítat podle vztahů

$$\begin{aligned}
&\frac{\partial I_j}{\partial x_{il}^A} = \frac{\partial I_{jl}}{\partial x_{il}^A}, \quad \frac{\partial I_j}{\partial x_{il}^B} = \frac{\partial I_{jl}}{\partial x_{il}^B}, \quad \frac{\partial I_j}{\partial x_{il}^S} = \frac{\partial I_{jl}}{\partial x_{il}^S}, \quad l \in L_j, \\
&\frac{\partial I_j}{\partial x_{il}^A} = 0, \qquad \frac{\partial I_j}{\partial x_{il}^A} = 0, \qquad \frac{\partial I_j}{\partial x_{il}^A} = 0, \qquad l \notin L_j,
\end{aligned}$$

kam dosazujeme hodnoty uvedené v předchozím oddílu. Nechť $\overline{I}$ je předepsaná hodnota rovnoměrného osvitu (společná pro všechny prvky formy). Pak účelovou funkci definujeme jako součet čtverců

$$F(x) = \frac{1}{2} \sum_{j=1}^{n_e} (I_j - \overline{I})^2,$$

kde vektor x má složky x_{1l}^A, x_{2l}^A, x_{3l}^A, x_{1l}^B, x_{2l}^B, x_{3l}^B, x_{1l}^S, x_{2l}^S, x_{3l}^S, $1 \leq l \leq n_l$ (devět pro každou lampu). Pak

$$\frac{\partial F(x)}{\partial x_{il}^A} = \sum_{j=1}^{n_e} (I_j - \overline{I}) \frac{\partial I_j}{\partial x_{il}^A}, \quad \frac{\partial F(x)}{\partial x_{il}^B} = \sum_{j=1}^{n_e} (I_j - \overline{I}) \frac{\partial I_j}{\partial x_{il}^B}, \quad \frac{\partial F(x)}{\partial x_{il}^S} = \sum_{j=1}^{n_e} (I_j - \overline{I}) \frac{\partial I_j}{\partial x_{il}^S}$$

kam dosazujeme derivace vypočtené z předchozích vztahů. Předepsanou hodnotu rovnoměrného osvitu vypočteme před zahájením optimalizačního procesu jako průměrnou hodnotu

$$\overline{I} = \frac{1}{n_e} \sum_{j=1}^{n_e} I_j.$$

Jak bylo uvedeno v předchozím oddílu, minimalizujeme funkci $F(x)$ na množině zadané omezeními ve tvaru rovností tvaru

$$\begin{aligned}
c_{1l}(x) &= \sum_{i=1}^{3} (x_{il}^S)^2 = 1, \\
c_{2l}(x) &= \sum_{i=1}^{3} x_{il}^S (x_{il}^B - x_{il}^A) = 0, \\
c_{3l}(x) &= \sum_{i=1}^{3} (x_{il}^B - x_{il}^A)^2 = d^2,
\end{aligned}$$

kde $1 \le l \le n_l$ (tři pro každou lampu). Platí

$$\frac{\partial c_{1l}(x)}{\partial x_{il}^A} = 0, \qquad \frac{\partial c_{1l}(x)}{\partial x_{il}^B} = 0, \qquad \frac{\partial c_{1l}(x)}{\partial x_{il}^S} = 2x_{il}^S,$$

$$\frac{\partial c_{2l}(x)}{\partial x_{il}^A} = -x_{il}^S, \qquad \frac{\partial c_{2l}(x)}{\partial x_{il}^B} = x_{il}^S, \qquad \frac{\partial c_{2l}(x)}{\partial x_{il}^S} = x_{il}^B - x_{il}^A,$$

$$\frac{\partial c_{2l}(x)}{\partial x_{il}^A} = -2(x_{il}^B - x_{il}^A), \quad \frac{\partial c_{2l}(x)}{\partial x_{il}^B} = 2(x_{il}^B - x_{il}^A), \quad \frac{\partial c_{2l}(x)}{\partial x_{il}^S} = 0$$

a ostatní derivace jsou nulové. Vzhledem k tomu, že každé omezení obsahuje pouze proměnné odpovídající jedné lampě, jsou omezení řídká a práce s nimi není náročná na použitou paměť ani na počet aritmetických operací.

Shrneme-li vlastnosti optimalizační úlohy, jde o minimalizaci součtu čtverců s nelineárními omezeními ve tvaru rovností. Počet dílčích funkcí v součtu čtverců je n_e (počet elementů formy). Počet proměnných, jejichž hodnoty se určují, je $9n_l$ (devět na každou lampu). Hessova matice účelové funkce není řídká. Počet nelineárních omezení je $3n_l$ (tři na každou lampu). Jacobiova matice nelineárních omezení je řídká. Tento výčet podmiňuje výběr optimalizační metody popsané ve třetí části práce.

Uvedeme ještě hrubý postup výpočtu. Nejprve se načtou vstupní údaje, kterými jsou vektory $x_j^T = (x_{1j}^T, x_{1j}^T, x_{1j}^T)$, $x_j^N = (x_{1j}^N, x_{1j}^N, x_{1j}^N)$, $1 \le j \le n_e$, popisující elementy formy, vektory $x_l^A = (x_{1l}^A, x_{1l}^A, x_{1l}^A)$, $x_l^B = (x_{1l}^B, x_{1l}^B, x_{1l}^B)$, $x_l^S = (x_{1l}^S, x_{1l}^S, x_{1l}^S)$, $1 \le l \le n_l$, určující počáteční nastavení lamp a údaje o vzájemném propojení elementů a lamp (pole IL a JL). Pak se vyvolá podprogram INITP, který provede normalizaci $v_j = -x_j^N/\|x_j^N\|$, $1 \le j \le n_e$ (takže $\|v_j\| = 1$ a $\tilde{v}_j = v_j$, $1 \le j \le n_e$), a podprogram CONST, který stanoví strukturu řídkosti Jacobiovy matice omezujících funkcí (pole ICG a JCG). Dále se určí délka lampy d, zvolí se počet svítících bodů p a vyvolá se podprogram INITK, který vypočte pomocné hodnoty d_k, λ_k, $1 - \lambda_k$, $1 \le k \le p$ (předpokládáme, že všechny lampy jsou stejné). Nakonec se spustí optimalizační program, který vyvolává procedury pro výpočet účelové funkce a omezujících funkcí. Výpočet účelové funkce začíná tím, že se vyvolá podprogram INITL, který znormalizuje vektory směru svitu lamp. Tím se spočtou normy $\|u_l\| = \|x_l^S\|$ a vektory $\tilde{u}_l = x_l^S/\|x_l^S\|$ pro $1 \le l \le n_l$. Pak následuje cyklus pro všechny elementy formy (pro $1 \le j \le n_e$). V tomto cyklu se nejprve vyvolá podprogram COMPL, ve kterém se pro každou lampu s indexem $j \in L_j$ vypočte (a uloží do paměti) intenzita osvitu I_{jl} a její derivace (podle vzorců uvedených v předchozím oddílu). Tyto veličiny se pak použijí pro výpočet hodnoty a gradientu účelové funkce. Výpočet hodnot a gradientů omezujících funkcí provádí podprogram CONSL. Používají se přitom pole ICG a JCG definující strukturu řídkosti.

3 Výběr optimalizační metody

Metoda rekursivního kvadratického programování

Úlohou, kterou máme řešit, je nalezení minima dvakrát spojitě diferencovatelné funkce $F : R^n \to R$, na množině zadané omezeními ve tvaru rovností

$$c_i(x) = 0, \qquad 1 \le i \le m.$$

Zde $x \in R^n$ je vektor n proměnných a $c_i : R^n \to R$, $1 \le i \le m \le n$, jsou dvakrát spojitě diferencovatelné funkce. Jsou-li splněny podmínky regularity (lineární nezávislost gradientů omezujících funkcí) mají nutné podmínky pro extrém tvar

$$\begin{aligned} \nabla F(x) + A(x)u &= 0, \\ c(x) &= 0. \end{aligned}$$

Je to $n + m$ nelineárních rovnic pro neznámé vektory $x \in R^n$ a $u \in R^m$, kde $A(x)$ je Jacobiova matice zobrazení $c(x)$ a u je vektor Lagrangeových multiplikátorů.

K řešení dané úlohy použijeme metodu rekursivního kvadratického programování, jejíž základní princip spočívá v aplikaci Newtonovy metody na soustavu nelineárních rovnic vyjadřující nutné podmínky pro extrém. Krok Newtonovy metody má tvar

$$\begin{aligned} x_{k+1} &= x_k + \alpha_k d_k^x, \\ u_{k+1} &= u_k + \alpha_k d_k^u, \end{aligned}$$

kde d_k^x, d_k^u jsou směrové vektory získané řešením soustavy lineárních rovnic

$$\begin{bmatrix} G(x_k, u_k) & A(x_k) \\ A(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d_k^x \\ d_k^u \end{bmatrix} = - \begin{bmatrix} g(x_k, u_k) \\ c(x_k) \end{bmatrix}$$

(lineární KKT systém) a $\alpha_k > 0$ je délka kroku. Zde

$$g(x, u) = \nabla F(x) + \sum_{i=1}^{m} u_i \nabla c_i(x) \quad \text{a} \quad G(x, u) = \nabla^2 F(x) + \sum_{i=1}^{m} u_i \nabla^2 c_i(x)$$

jsou po řadě gradient a Hessova matice Lagrangeovy funckce $L(x, u) = F(x) + u^T c(x)$. V našem případě nebudeme používat přímo Hessovu matici Lagrangeovy funkce, ale její aproximaci B_k získanou pomocí metody BFGS s proměnnou metrikou. Pak

$$\begin{bmatrix} B_k & A_k \\ A_k^T & 0 \end{bmatrix} \begin{bmatrix} d_k^x \\ d_k^u \end{bmatrix} = - \begin{bmatrix} g(x_k, u_k) \\ c(x_k) \end{bmatrix},$$

kde $B_1 = I$ (I je jednotková matice řádu n) a

$$B_{k+1} = B_k + \frac{1}{s_k^T y_k} y_k y_k^T - \frac{1}{s_k^T B_k s_k} (B_k s_k y_k^T + y_k s_k^T B_k),$$

kde $s_k = x_{k+1} - x_k = \alpha_k d_k$ a $y_k = g(x_{k+1}, u_{k+1}) - g(x_k, u_{k+1})$.

Pro výběr délky kroku α_k se používají různé pokutové funkce. V našem případě použijeme rozšířenou Lagrangeovu funkci

$$P_k(\alpha) = F(x_k + \alpha d_k^x) + (u_k + d_k^u)^T c(x_k + \alpha d_k^x) + \frac{\sigma}{2}\|c(x_k + \alpha d_k^x)\|^2,$$

kde $\sigma \geq 0$ je pokutový parametr. V práci [2] je ukázáno, že řešíme-li lineární KKT systém tak přesně, že

$$\|G_k d_k^x + A_k d_k^u + g_k\| \leq \overline{\omega}_k \|g_k\| \quad \text{a} \quad \|A_k^T d_k^x + c_k\| \leq \overline{\omega}_k \|c_k\|,$$

kde $0 < \overline{\omega}_k < 1$, platí $P_k'(0) < 0$ (první derivace funkce $P_k(\alpha)$ je pro $\alpha = 0$ záporná) a funkce $P_k(\alpha)$ klesá ve směru d_k^x. Jelikož $P_k'(0) < 0$, můžeme délku kroku volit tak, že $\alpha_k = \beta^{j-1} \max(1, \overline{\Delta}/\|d_k^x\|)$, kde $\overline{\Delta}$ je maximální délka kroku, $0 < \beta < 1$ je koeficient redukce a $j \in N$ je nejmenší přirozené číslo takové, že

$$P_k(\alpha_k) - P_k(0) \leq \varepsilon_1 \alpha_k P_k'(0),$$

kde $0 < \varepsilon_1 < 1/2$ je Armijův parametr. Používají se hodnoty $\overline{\Delta} = 1000$, $\beta = 0.5$ a $\varepsilon_1 = 0.0001$.

Řešení lineárního KKT systému

V tomto oddílu budeme vynechávat index iteračního kroku k a lineární KKT systém zapíšeme ve tvaru

$$Kd = \begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} d^x \\ d^u \end{bmatrix} = \begin{bmatrix} b^x \\ b^u \end{bmatrix} = b. \tag{3.1}$$

Tato symetrická soustava lineárních rovnic, jejíž matice je indefinitní, se řeší předpodmíněnou metodou sdružených gradientů s indefinitním přefpodmiňovačem

$$C = \begin{bmatrix} D & A \\ A^T & 0 \end{bmatrix},$$

kde D je pozitivně definitní diagonální matice odvozená z diagonály matice B. Volba tohoto předpodmiňovače je zdůvodněna v [2]. Násobení vektoru r maticí C^{-1} lze vyjádřit ve tvaru

$$\begin{aligned} C^{-1}r &= \begin{bmatrix} D^{-1} - D^{-1}A(A^TD^{-1}A)^{-1}A^TD^{-1} & D^{-1}A(A^TD^{-1}A)^{-1} \\ (A^TD^{-1}A)^{-1}A^TD^{-1} & -(A^TD^{-1}A)^{-1} \end{bmatrix} \begin{bmatrix} r^x \\ r^u \end{bmatrix} \\ &= \begin{bmatrix} D^{-1}(r^x - At^u) \\ t^u \end{bmatrix}, \qquad t^u = (A^TD^{-1}A)^{-1}(A^TD^{-1}r^x - r^u). \end{aligned}$$

Předpodmíněnou metodu sdružených gradientů lze popsat takto. Nejprve položíme $d_1 = 0$, $r_1 = b$,

$$t_1^u = (A^TD^{-1}A)^{-1}(A^TD^{-1}r_1^x - r_1^u), \qquad t_1^x = D^{-1}(r_1^x - At_1^u)$$

a $p_1 = t_1$. Pak pro $i = 1, 2, 3, \ldots$ provádíme následující kroky. Jestliže $\|r_i^x\| \leq \overline{\omega}\|b^x\|$ a $\|r_i^u\| \leq \overline{\omega}\|b^u\|$, kde $\overline{\omega}$ je zadaná přesnost, položíme $d = d_i$ a ukončíme výpočet. V opačném případě vypočteme

$$\begin{aligned}
q_i &= Kp_i, \qquad \alpha_i = r_i^T t_i / p_i^T q_i, \\
d_{i+1} &= d_i + \alpha_i p_i, \qquad r_{i+1} = r_i - \alpha_i q_i, \\
t_{i+1}^u &= (A^T D^{-1} A)^{-1} (A^T D^{-1} r_{i+1}^x - r_{i+1}^u), \\
t_{i+1}^x &= D^{-1}(r_{i+1}^x - A t_{i+1}^u), \\
\beta_i &= r_{i+1}^T t_{i+1} / r_i^T t_i, \qquad p_{i+1} = t_{i+1} + \beta_i p_i.
\end{aligned}$$

V těchto vztazích používáme vektory

$$r = \begin{bmatrix} r^x \\ r^u \end{bmatrix}, \quad t = \begin{bmatrix} t^x \\ t^u \end{bmatrix}.$$

V uvedeném popisu se vyskytuje násobení vektoru maticí $(A^T D^{-1} A)^{-1}$. Tuto matici není třeba počítat explicitně, používá se místo ní Choleského rozklad $LL^T = A^T D^{-1} A$, kde L je dolní trojúhelníková matice.

Použití systému UFO

K optimalizaci osvitu formy lze použít systém pro univerzální funkcionální optimalizaci UFO [1]. Tento systém obsahuje velké množství optimalizačních metod pro řešení různých typů optimalizačních úloh, mezi jinými také metody sekvenčního kvadratického programování pro úlohy s nelineárními omezeními ve tvaru rovností, používající aktualizace s proměnnou metrikou pro aproximaci Hessovy matice Lagrangeovy funkce (volba $FORM='SE', $CLASS='VM'). Systém UFO je ovládán vstupním jazykem. Příkazy vstupního jazyka se zadávají pomocí vstupní šablony, jejíž forma pro naší úlohu je uvedena v poslední části práce. Šablona obsahuje zadání vstupních údajů a inicializaci (makroproměnná $INPUT), tisk výstupních údajů (makroproměnná $OUTPUT), zadání hodnoty a gradientu účelové funkce (makroproměnná $FGMODELF), zadání hodnot a gradientů omezujících funkcí (makroproměnná $FGMODELC), připojení externích podprogramů (makroproměnná $SUBROUTINES), velikost optimalizační úlohy ($NF, $NA, $NC), typ úlohy ($MODEL='FF', $JACC='S', $HESF='D'), specifikaci úlohy ($FORM='SE', $CLASS='VM') a další potřebné informace (vše je podrobně popsáno v [1]).

4 Datové soubory pro optimalizaci osvitu

V této části je ukázáno, jak se zadávají elementy formy a parametry lamp. Dále jsou uvedeny vstupní údaje odpovídající testovacím úlohám. Vstupními datovými soubory optimalizace jsou soubor elementů, soubor lamp, soubor topologie. Ve všech případech se jedná o textové soubory obsahující pouze číselné hodnoty navzájem od sebe oddělené minimálně jedním „prázdným znakem“ tj. mezerou, tabulátorem nebo koncem řádku. Číselné hodnoty v jednotlivých souborech mají následující význam.

Soubor elementů:

Soubor elementů má zpravidla příponu ELM. Data každého elementu jsou zaznamenána na jednom řádku souboru. Každý řádek souboru obsahuje parametry elementu v pořadí j, x^T_{1j}, x^T_{2j}, x^T_{3j}, x^N_{1j}, x^N_{2j}, x^N_{3j}, kde $0 \leq j \leq n_e$ je index elementu.

Soubor lamp:

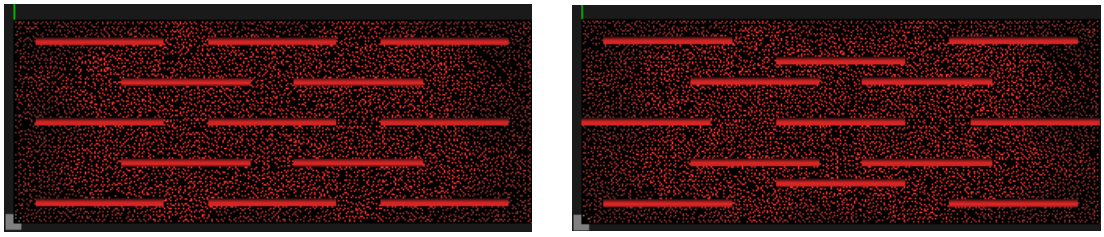
Soubor lamp má zpravidla příponu LMP. Každý řádek souboru obsahuje parametry jedné lampy v pořadí l, x^A_{1l}, x^A_{2l}, x^A_{3l}, x^B_{1l}, x^B_{2l}, x^B_{3l}, x^S_{1l}, x^S_{2l}, x^S_{3l}, kde $0 \leq l \leq n_l$ je index lampy.

Soubor topologie:

Soubor topologie má zpravidla příponu TOP. Na každém řádku souboru je dvojice hodnot l a j, vyjadřující skutečnost, že element j je potenciálně ohříván lampou l.

Sady souborů

Pro první testování byly připraveny sady souborů, kde plochou formy je rovina s nulovou třetí složkou. Plocha formy má rozměry 120 $\times$ 50 cm a je rozdělena zhruba na jedenáct tisíc elementů. Jednotlivé úlohy se liší počáteční pozicí lamp.



Obrázek 4.1: Ukázka počátečního rozmístění lam nad definovanou plochou

5 Výsledky testovacích úloh

V první fázi jsme vybrali úlohy, ve kterých bylo požadováno optimální rozmístění lamp nad rovinnou plochou.

Plocha a síť

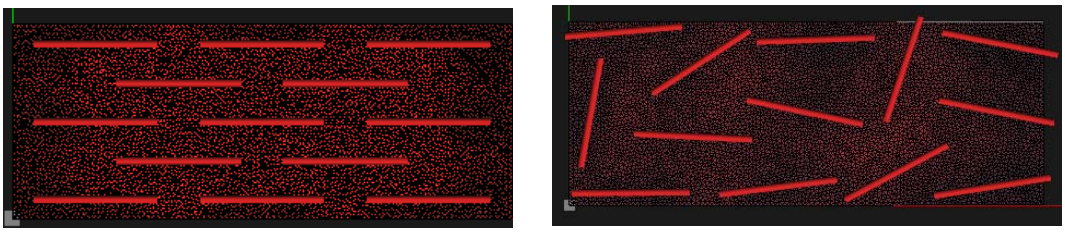
Pro výpočet prvních testovacích úloh byla zvolena rovinná plocha velikosti 120 $\times$ 50 cm. Plocha byla pokryta sítí trojúhelníkových elementů v počtu 11793.

Lampy

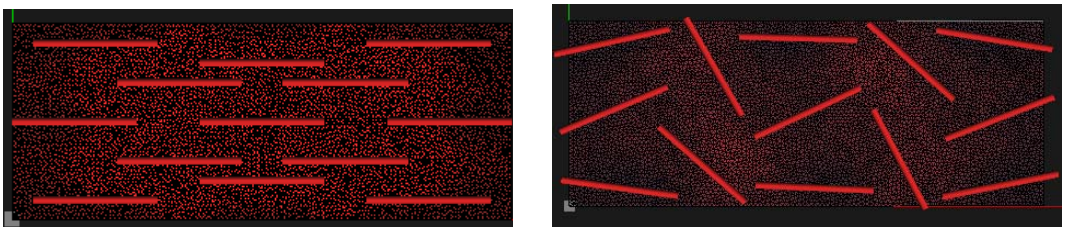
Nad plochou byly rozmístěny lampy v celkovém počtu 13. Prozatím byly uvažovány pouze případy, kdy všechny lampy mají shodnou délku 30 cm. Počáteční vzdálenost lamp od osvětlované plochy byla ve všech případech volena 10 cm.

Výsledky

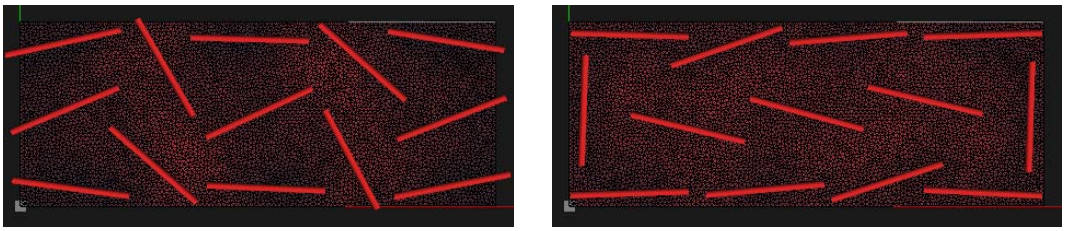
V první úloze ULOHA11 byla počáteční pozice lamp zahřívajících plochu zvolena víceméně rovnoměrně nad definovanou plochou (viz obrázek 5.1 vlevo), směrový úhel záření lamp je kolmý na zahřívanou plochu. Výsledné rozmístění lamp získané optimalizací je zobrazeno na obrázku 5.1 vpravo, směrový úhel záření lamp zůstává po optimalizaci prakticky shodný s počátečním směrovým úhlem (odchylky od původního směru kolmého na plochu jsou do 1%). Na základě získaných výsledků byla připravena druhá iterace tohoto výpočtu jako ULOHA11_2, kde počáteční rozmístění lamp nad plochou bylo definováno dle výsledného řešení pro ULOHA11. Počáteční a výsledná pozice lamp (viz obrázek 5.2) v tomto případě nevykazují výraznější odchylky. Dále byla řešena úloha ULOHA12 s počátečním rozmístěním lamp dle obrázku 5.3 vlevo. Výsledné rozmístění vypočtené optimalizačním kódem je zobrazené na obrázku 5.3 vpravo.



Obrázek 5.1: Počáteční pozice lamp (vlevo) a pozice lamp na plochou po optimalizaci (vpravo) pro výpočet ULOHA11.



Obrázek 5.2: Počáteční pozice lamp (vlevo) a pozice lamp na plochou po optimalizaci (vpravo) pro výpočet ULOHA11_2 (druhá iterace výpočtu ULOHA11).



Obrázek 5.3: Počáteční pozice lamp (vlevo) a pozice lamp na plochou po optimalizaci (vpravo) pro výpočet ULOHA12.

Zhodnocení

V první fázi testování byly připraveny základní jednoduché úlohy, z nichž některé byly uvedeny v předchozím textu. Na základě dosažených výsledků lze konstatovat, že pro výše definované různé počáteční pozice bylo dosaženo prakticky stejné výsledné optimalizované polohy lamp. Po optimalizaci jsou lampy do jisté míry "rozházené" nad plochou, což může způsobit, že některé lampy ve výsledku zahřívají i část plochy, která nebyla původně pro danou lampu zaznamenána v souboru topologie. V tomto případě by optimalizační algoritmus pracoval s nepřesnou informací a jeho výsledky by byly zkreslené. Z toho důvodu byla pro ULOHA11 provedena druhá iterace - v tomto případě nebyla ve výsledku zaznamenána praktická změna výsledné pozice lamp oproti iteraci první. Na základě získaných výsledků lze pro další fázi vývoje doporučit do optimalizačního algoritmu zahrnout omezující kritéria, která by zamezila případným příliš velkým změnám optimalizovaných pozičních parametrů lamp od počátečních tak, aby v případě řešení složitějších úloh nedocházelo k posunu lamp mimo plochu nebo do velké vzdálenosti od plochy.

6 Výpisy programů

Vstupní šablona pro systém UFO

```
$SET(INPUT)
  MA=0
  NA=0
  NL=0
  OPEN(11,FILE='LAMPA1.ELM',STATUS='UNKNOWN')
  OPEN(12,FILE='LAMPA1.LMP',STATUS='UNKNOWN')
  OPEN(13,FILE='LAMPA1.TOP',STATUS='UNKNOWN')
  OPEN(14,FILE='LAMPA1.OUT',STATUS='UNKNOWN')
1 READ(11,*,END=2) KA,(XT(3*(KA-1)+I),I=1,3),(VT(3*(KA-1)+I),I=1,3)
  NA=MAX(KA,NA)
  GO TO 1
2 READ(12,*,END=3) L,(X(9*(L-1)+I),I=1,3),(X(9*(L-1)+I+3),I=1,3)&
,(X(9*(L-1)+I+6),I=1,3)
  NL=MAX(L,NL)
  GO TO 2
3 MA=MA+1
  READ(13,*,END=4) JL(MA),IL(MA)
  GO TO 3
4 CONTINUE
  MA=MA-1
  NF=9*NL
  NC=3*NL
  CALL UASED3(NA,MA,IL,JL)
  NP=20
  HN=SQRT((X(4)-X(1))**2+(X(5)-X(2))**2+(X(6)-X(3))**2)
  DO KA=1,NA
  CALL INITP(VT(3*(KA-1)+1))
  END DO
  CALL INITK(NP,HD,HN,OLA,RLA)
  DO L=1,NL
  DO I=1,3
  IC(3*(L-1)+I)=5
  CL(3*(L-1)+I)=0.0D0
  END DO
  CALL CSTRL(ICG(3*(L-1)+1),JCG(18*(L-1)+1),L)
  END DO
  ICG(3*NL+1)=18*NL+1
  X(9*NL+1)=0.0D0
  DO L=1,NL
  CALL INITL(X(9*(L-1)+7),U(3*(L-1)+1),UN(L))
  END DO
```

```
  DO KA=1,NA
  FA=0.0D0
  ISTRT=IL(KA)
  ISTOP=IL(KA+1)-1
  DO J=ISTRT,ISTOP
  L=JL(J)
  CALL COMPL(X(9*(L-1)+1),X(9*(L-1)+4),U(3*(L-1)+1),UN(L)&
,XT(3*(KA-1)+1),VT(3*(KA-1)+1),NP,HD,OLA,RLA,FA,GL(9*(L-1)+1)&
,GL(9*(L-1)+4),GL(9*(L-1)+7),0)
  END DO
  X(9*NL+1)=X(9*NL+1)+FA
  END DO
  X(9*NL+1)=X(9*NL+1)/DBLE(NA)
$ENDSET
$SET(FGMODELF)
  DO L=1,NL
  CALL INITL(X(9*(L-1)+7),U(3*(L-1)+1),UN(L))
  END DO
  FF=0.0D0
  CALL UXVSET(NF,0.0D0,GF)
  DO KA=1,NA
  FA=0.0D0
  ISTRT=IL(KA)
  ISTOP=IL(KA+1)-1
  DO J=ISTRT,ISTOP
  L=JL(J)
  CALL COMPL(X(9*(L-1)+1),X(9*(L-1)+4),U(3*(L-1)+1),UN(L)&
,XT(3*(KA-1)+1),VT(3*(KA-1)+1),NP,HD,OLA,RLA,FA,GL(9*(L-1)+1)&
,GL(9*(L-1)+4),GL(9*(L-1)+7),KD)
  END DO
  FA=FA-X(9*NL+1)
  GF(9*NL+1)=GF(9*NL+1)-FA
  DO J=ISTRT,ISTOP
  L=JL(J)
  DO I=1,3
  GF(9*(L-1)+I)=GF(9*(L-1)+I)+FA*GL(9*(L-1)+I)
  GF(9*(L-1)+I+3)=GF(9*(L-1)+I+3)+FA*GL(9*(L-1)+I+3)
  GF(9*(L-1)+I+6)=GF(9*(L-1)+I+6)+FA*GL(9*(L-1)+I+6)
  END DO
  END DO
  FF=FF+FA**2
  END DO
  FF=0.5D0*FF
$ENDSET
```

```
$SET(FGMODELCS)
  DO L=1,NL
  CALL CONSL(X(9*(L-1)+1),X(9*(L-1)+4),X(9*(L-1)+7),CF(3*(L-1)+1)&
,CG(18*(L-1)+1),HN,KD)
  END DO
$ENDSET
$SET(OUTPUT)
  DO L=1,NL
  WRITE(14,'(I3,1X,9D15.7)') L,(X(9*(L-1)+I),I=1,3)&
,(X(9*(L-1)+I+3),I=1,3),(X(9*(L-1)+I+6),I=1,3)
  END DO
$ENDSET
$ADD(SUBROUTINES)
  $INCLUDE('LAMPY\COMPL')
  $INCLUDE('LAMPY\CONSL')
  $INCLUDE('LAMPY\CSTRL')
  $INCLUDE('LAMPY\INITK')
  $INCLUDE('LAMPY\INITL')
  $INCLUDE('LAMPY\INITP')
$ENDADD
$MOUT=2
$NOUT=2
$NF=1000
$NA=100000
$NC=5000
$MA=100000
$MC=15000
$M=250000
$NL=100
$NP=20
$ADD(INTEGER,'\IL($MA+$NL)\JL($MA)\NL\NP\ISTRT\ISTOP')
$ADD(REAL,'\VT(3*$NA)\XT(3*$NA)\U(3*$NL)\UN($NL)')
$ADD(REAL,'\OLA($NP)\RLA($NP)\HD($NP)\HN\GL($NF)')
$MODEL='FF'
$JACC='S'
$HESF='D'
$FORM='SE'
$CLASS='VM'
$REM $TEST='YES'
$TOLG='1.0D-5'
$MIT=20000
$MFV=20000
$MFG=20000
$BATCH
$STANDARD
```

Použité procedury

```
SUBROUTINE INITP(V)
REAL*8 V(3)
INTEGER I
REAL*8 VN
VN=0.0D0
DO I=1,3
VN=VN+V(I)**2
END DO
VN=SQRT(VN)
DO I=1,3
V(I)=-V(I)/VN
END DO
RETURN
END

SUBROUTINE INITK(P,H,HN,OLA,RLA)
INTEGER P
REAL*8 H(P),HN,OLA(P),RLA(P)
INTEGER K
DO K=1,P
RLA(K)=DBLE(K-1)/DBLE(P-1)
OLA(K)=1.0D0-RLA(K)
H(K)=HN/DBLE(P-1)
END DO
H(1)=0.5D0*H(1)
H(P)=0.5D0*H(P)
RETURN
END

SUBROUTINE INITL(XU,U,UN)
REAL*8 XU(3),U(3),UN
INTEGER I
UN=0.0D0
DO I=1,3
UN=UN+XU(I)**2
END DO
UN=SQRT(UN)
DO I=1,3
U(I)=XU(I)/UN
END DO
RETURN
END
```

```
 SUBROUTINE CSTRL(ICG,JCG,L)
 INTEGER ICG(3),JCG(18),L
 INTEGER I,M,N
 M=18*(L-1)
 N= 9*(L-1)
 ICG(1)=M+1
 ICG(2)=M+7
 ICG(3)=M+16
 DO I=1,3
 JCG(I)=N+I
 JCG(I+3)=N+I+3
 JCG(I+6)=N+I
 JCG(I+9)=N+I+3
 JCG(I+12)=N+I+6
 JCG(I+15)=N+I+6
 END DO
 RETURN
 END

 SUBROUTINE COMPL(XA,XB,U,UN,XT,V,P,H,OLA,RLA,RI,DIXA,DIXB,DIXU,KD)
 INTEGER P,KD
 REAL*8 XA(3),XB(3),U(3),UN,XT(3),V(3),H(P),OLA(P),RLA(P),RI,
& DIXA(3),DIXB(3),DIXU(3)
 INTEGER I,K
 REAL*8 W(3),WW,WN,ALF,BET,ROT,Z1,Z2,Z3,DIWI
 DO I=1,3
 DIXA(I)=0.0D0
 DIXB(I)=0.0D0
 DIXU(I)=0.0D0
 END DO
 DO K=1,P
 WW=0.0D0
 DO I=1,3
 W(I)=OLA(K)*XA(I)+RLA(K)*XB(I)-XT(I)
 WW=WW+W(I)**2
 END DO
 WN=SQRT(WW)
 ALF=0.0D0
 BET=0.0D0
 DO I=1,3
 W(I)=-W(I)/WN
 ALF=ALF+U(I)*W(I)
 BET=BET+V(I)*W(I)
 END DO
 ROT=SQRT(1.0D0-ALF**2)
```

```
Z1=(3.0D 0*ALF+0.5D 0*ROT)*H(K)/WW
RI=RI+Z1*BET
IF (KD.GT.0) THEN
Z1=Z1/WN
Z2=(3.0D 0-0.5D 0*ALF/ROT)*BET*H(K)/WW
Z3=Z2/UN
Z2=Z2/WN
DO I=1,3
DIXU(I)=DIXU(I)+Z3*(W(I)-ALF*U(I))
DIWI=Z2*(U(I)-ALF*W(I))+Z1*(V(I)-3.0D0*BET*W(I))
DIXA(I)=DIXA(I)-OLA(K)*DIWI
DIXB(I)=DIXB(I)-RLA(K)*DIWI
END DO
END IF
END DO
RETURN
END

SUBROUTINE CONSL(XA,XB,XU,CF,CG,HN,KD)
INTEGER KD
REAL*8 XA(3),XB(3),XU(3),CF(3),CG(18),HN
INTEGER I
REAL*8 DIF
CF(1)=-HN**2
CF(2)= 0.0D0
CF(3)=-1.0D0
DO I=1,3
DIF=XB(I)-XA(I)
CF(1)=CF(1)+DIF**2
CF(2)=CF(2)+DIF*XU(I)
CF(3)=CF(3)+XU(I)**2
END DO
IF (KD.GT.0) THEN
DO I=1,3
DIF=XB(I)-XA(I)
CG(I)=-2.0D0*DIF
CG(I+3)=2.0D0*DIF
CG(I+6)=-XU(I)
CG(I+9)= XU(I)
CG(I+12)=DIF
CG(I+15)=2.0D0*XU(I)
END DO
END IF
RETURN
END
```

Literatura

[1] L.Lukšan, M.Tůma, J.Vlček, N.Ramešová, M.Šiška, J.Hartman, C.Matonoha: UFO 2008 - Interactive system for universal functional optimization. Technical Report V-1040. Institute of Computer Science, Academy of Sciences, Prague 2008.

[2] L.Lukšan, J.Vlček: Indefinitely preconditioned inexact Newton method for large sparse equality constrained nonlinear programming problems. Numerical Linear Algebra with Applications, 5, 219-247, 1998.