

Řešení kontaktní úlohy v MKP s ohledem na efektivitu výpočtu

Jan Hynouš

Abstrakt

Tato práce se zabývá řešením kontaktní úlohy v MKP s ohledem na efektivitu výpočtu. Na její realizaci se spolupracovalo s firmou Ricardo Prague s. r. o., která působí v oblastech výzkumu a vývoje pro automobilový průmysl.

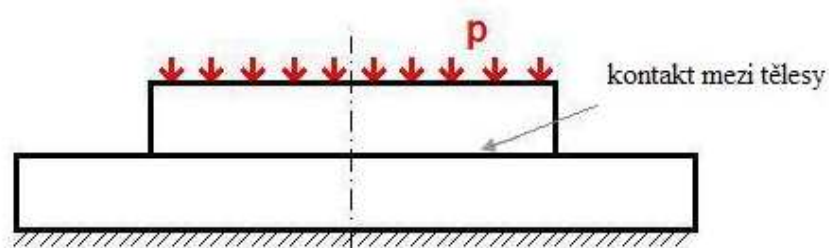
Práce pojednává o různých modelovacích technikách v konečně prvkovém softwaru Abaqus. Poznatky byly následně aplikovány na model ojnice. Na tomto modelu bylo ukázáno, že správně nastavený kontaktní problém společně s použitím dokonale tuhého tělesa a efektivním nastavením řešiče přinese zkrácení výpočtového času.

Klíčová slova

Kontaktní úloha, metoda konečných prvků, konvergence, Abaqus

1. Úvod

Kontaktní úloha popisuje interakci mezi tělesy z hlediska vzájemného silového působení a dodržení podmínek kompatibility a nepropustnosti. Na Obr. 1. je znázorněn příklad možného uspořádání kontaktní úlohy.



Obr. 1. Symetrická úloha s kontaktem mezi tělesy

1.1 Formulace kontaktu v programu Abaqus

Kontaktní úloha řeší problém interakce těles, které se navzájem dotýkají a působí na sebe silami. Kontakt se může definovat buďto jako kontakt dvou ploch, které působí vzájemně na sebe jako kontaktní pár, nebo jako self-kontakt, který interaguje sám se sebou.

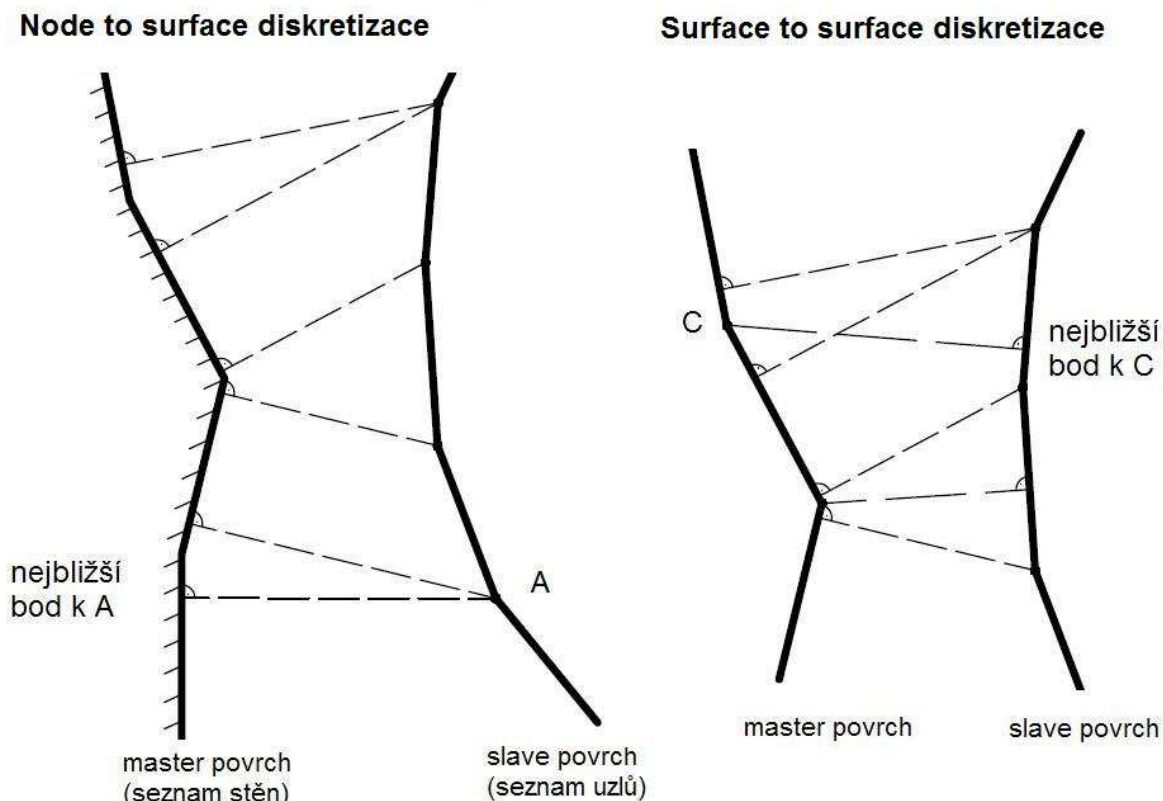
Kontakty se mohou použít při interakci tuhého tělesa s poddajným tělesem, stejně tak i při interakci mezi poddajnými tělesy. K definování kontaktu je potřeba vybrat plochy, které interagují mezi sebou. Abaqus poskytuje několik formulací kontaktů. Můžeme je dělit podle způsobu diskretizace kontaktních ploch a podle klasifikace vzájemného pohybu kontaktních ploch.

Klasifikace podle způsobu diskretizace kontaktních ploch:

Vazba uzlu ke stěně elementu (Node to surface)

Kontakt je ustanoven tak, že každý slave uzel na jedné kontaktní slave ploše interaguje se stěnou elementu na druhé kontaktní master ploše. Každá kontaktní podmínka zahrnuje jeden slave uzel a skupinu nejbližších master uzlů, z kterých jsou interpolovány hodnoty do konkrétních bodů. Vlastností tohoto algoritmu je, že slave uzly nemohou penetrovat do master

plochy. Naopak master povrch může penetrovat do slave plochy, což může v některých případech vést k nepřesnostem. Princip tohoto algoritmu je zobrazen na Obr. 2. vlevo.



Obr. 2. Vyhledávání kontaktních párů

Surface to surface

Tato diskretizace uvažuje tvary obou ploch (master i slave) v regionu vazbových podmínek. Kontaktní podmínky jsou uplatněny přes celou slave plochu. Významná penetrace se u tohoto algoritmu objevuje výjimečně a je případně rovnoměrně rozložena.

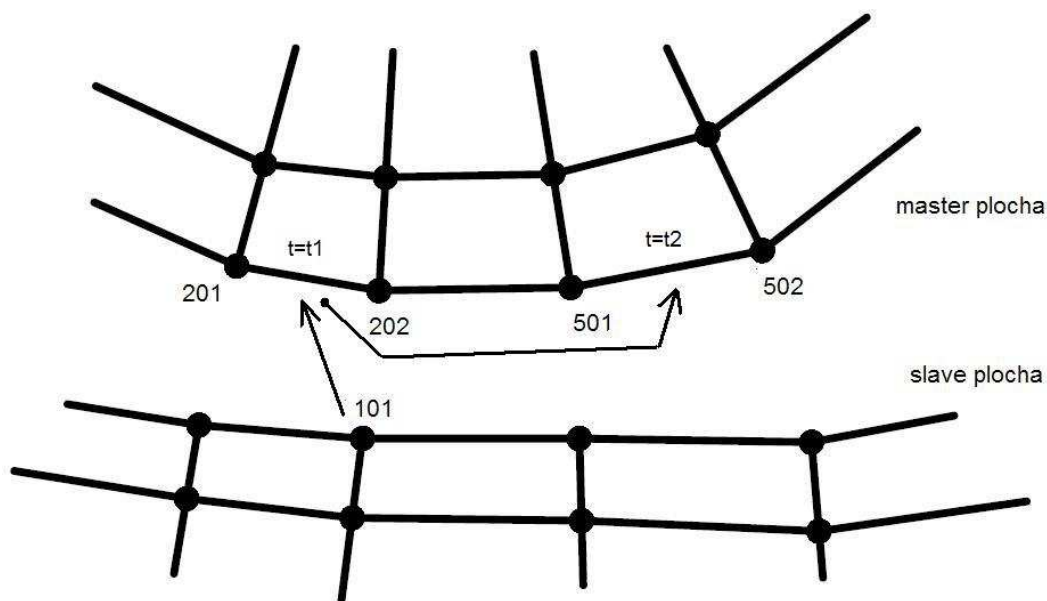
Surface to surface algoritmus poskytuje všobecně přesnější výsledky napětí, než algoritmus node to surface.

Přístup surface to surface poskytuje také hladší rozložení napětí než algoritmus node to surface, kde se mohou objevit lokální špičky napětí. Nicméně metoda surface to surface je náročnější na počítačový čas.

Klasifikace podle velikosti vzájemného pohybu kontaktních ploch:

Finite sliding

Tento přístup je nejobecnější a povoluje libovolnou separaci, relativní posunutí a rotaci ploch. Slave uzel se může libovolně pohybovat po master ploše a být s ní v kontaktu. Vyhledávání kontaktních párů proběhne v každém přírůstku zatížení. Jak je vidět z Obr 3. na počátku analýzy je slave uzel 101 v kontaktu například s elementem ohraničeným uzly 201 a 202, zatížení se bude přenášet mezi uzlem 101 a uzly 201 a 202. Později, díky deformaci povrchu se například dostane slave uzel 101 do kontaktu s elementem ohraničeným uzly 501 a 502. Pak se bude zatížení přenášet mezi uzlem 101 a uzly 501 a 502.

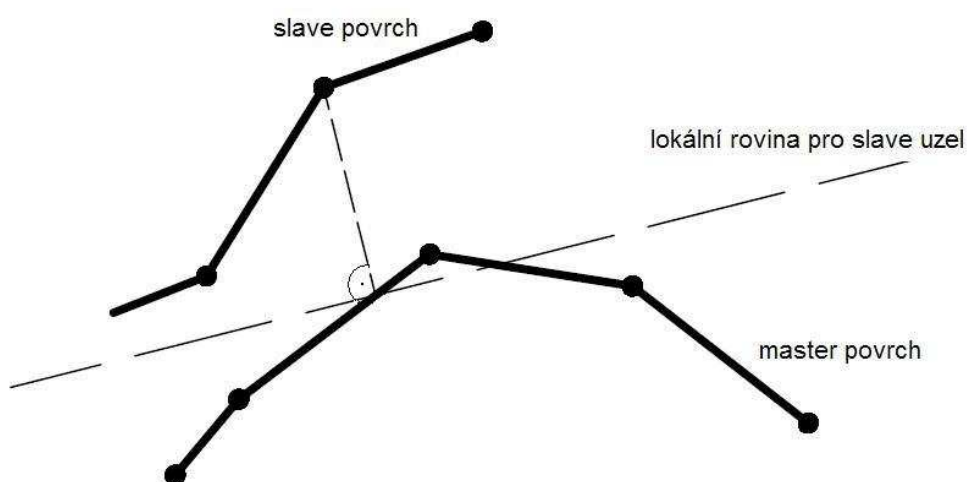


Obr. 3. Přístup finite sliding

Small sliding

Tento přístup vyžaduje relativně malé posouvání povrchů. Slave uzel interaguje během analýzy pořád se stejnou lokální stěnou master povrchu. Na začátku analýzy jsou vyhledány všechny kontaktní páry a definovány kontaktní roviny pro slave uzly. Metoda small sliding vyhledává kontaktní páry pouze na začátku analýzy. Princip této metody je znázorněn na Obr. 4.

Každý slave uzel má svoji kontaktní rovinu, která se během analýzy nemění. Nemusí se proto monitorovat celý master povrch. Lze tedy prohlásit, že tento přístup je výpočetně efektivnější než přístup finite sliding.



Obr. 4. Přístup small sliding

1.2 Dokonale tuhé těleso

Tuhé těleso je soubor uzlů, elementů a ploch, jehož pohyb je popsán pohybem jediného uzlu, který se nazývá referenční uzel tělesa. Referenční uzel má šest stupňů volnosti. Relativní pozice uzlů a ploch, které jsou částí dokonale tuhého tělesa zůstávají během simulace stejné. Proto se příslušné elementy nedeformují. Mohou se jen pohybovat.

Pohyb dokonale tuhého tělesa může být předepsán v referenčním uzlu tělesa. Tuhá tělesa mohou interagovat jak mezi sebou, tak i s poddajnými tělesy.

Tuhá tělesa mohou být použita pro velmi tuhé komponenty. Mohou být také použita k modelování vazeb mezi ideálními tělesy a poskytnout výhodnou metodu k určení kontaktní interakce. Dají se s výhodou aplikovat při zjednodušování modelů v místech, která nejsou z hlediska analýzy stěžejní.

Největší výhodou tuhých těles je jejich výpočetní nenáročnost. Výpočet není proveden pro elementy tohoto tělesa, pouze se přepočítá pohyb referenčního uzlu.

1.3 Řízení kontaktního výpočtu

Standardní řízení výpočtu v programu Abaqus je většinou dostatečné, nicméně přidáním dalších parametrů řízení výpočtu se dá docílit efektivnějšího řešení. To se dá především dosáhnout u modelů obsahující nelinearity jako jsou např. kontakty.

Řízení kontaktního výpočtu se provádí v Abaqusu pomocí příkazu `*CONTACT CONTROLS`, který má řadu parametrů. V této úloze byl použit parametr `STABILIZE`. Tento parametr aktivuje tlumení v normálovém i tečném směru, v závislosti na hustotě sítě kontaktních ploch a na velikosti časového kroku použitého při řešení. Tento parametr může být zadán buďto pro celý model nebo pro jednotlivé kontaktní páry. Pokud se nezadá hodnota koeficientu tlumení, Abaqus si vypočítá tlumící koeficient sám.

1.4 Možnosti řešících technik výpočtu

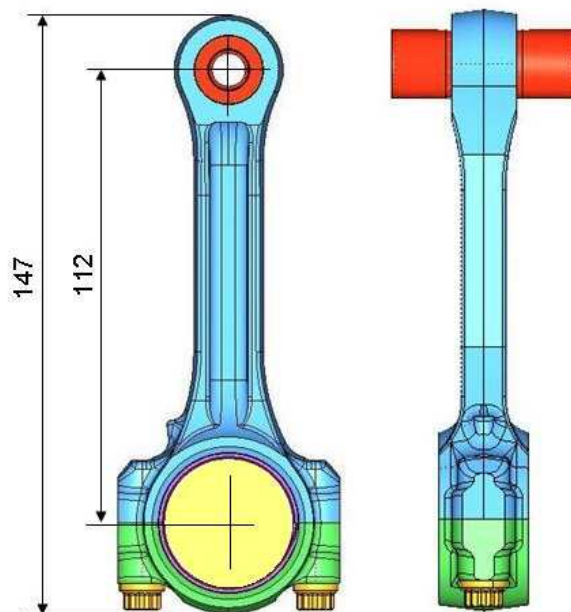
Abaqus používá k řešení MKP úlohy Newton-Raphsonovu metodu. Jako další metoda se dá použít Kvazi-Newtonova metoda. Ta se dá zejména použít pro velký systém, kde se matice tuhosti příliš nemění během iterací. Tato technika může výrazně ušetřit výpočtový čas.

Namísto klasického řešení kontaktů lze použít v Abaqusu řešící techniku `CONTACT ITERATIONS`. Použití této metody se dá zkrátit výpočtový čas. Je vhodná pro statické úlohy s více kontakty bez tření.

2. Aplikace na modelu ojnice

Výpočet byl proveden na modelu ojnice, která je zobrazena na Obr. 5. Simulace byla provedena v softwaru Abaqus. Bylo požadováno, aby se úpravy směřující k efektivnímu výpočtu zapsaly do input souboru, který Abaqus vytvoří po zkompletování modelu před výpočtem. Tento soubor je později možno upravovat v textovém editoru. Dají se sem vkládat příkazy, které se zapíšou ručně. Také je možnost upravovat již vložené hodnoty.

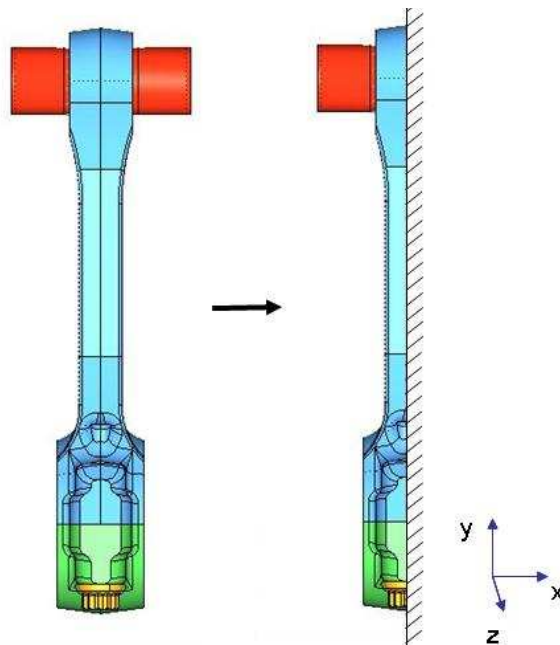
Sestava ojnice se skládá z ojnice, pístního čepu, víka ojnice, klikového čepu, dvou šroubů a vložky mezi ojnicí a klikovým čepem.



Obr. 5. Geometrie ojnice

2.1 Symetrie modelu

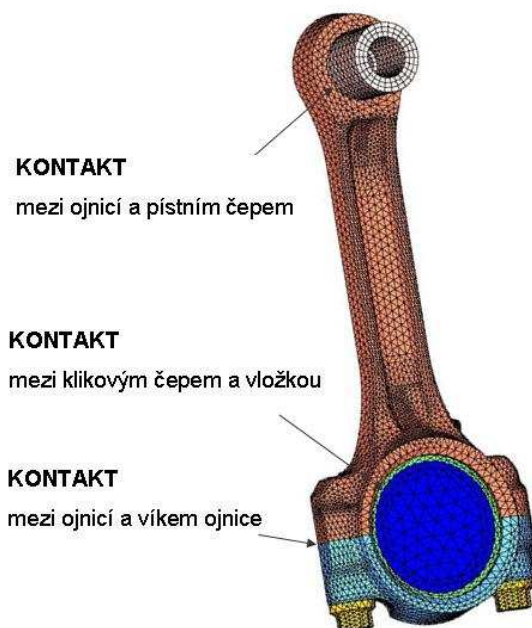
Úloha je symetrická podle roviny yz , lze modelovat pouze polovinu modelu. Následný výpočet modelu bude efektivnější zejména kvůli známým posuvům na rovině symetrie. Nutno podotknout, že v analýze nebylo potřeba řešit žádný výpočet, který by potřeboval celý model, jako např. dynamiku ojnice.



Obr. 6. Využití symetrie ojnice

2.2 Definice kontaktů

V úloze jsou použity kontakty zřejmé z Obr. 7. Kontakty jsou řešeny algoritmem master-slave s diskretizací ploch surface to surface a použitím přístupu small sliding.



Obr. 7. Definice kontaktů

2.3 Úprava input souboru před provedením výpočtu

Nastavení klikového čepu jako dokonale tuhé těleso

V modelu byl použit klikový čep nastaven jako dokonale tuhé těleso. Toto zjednodušení zajistí urychlení výpočtového času analýzy a lepší konvergenci kontaktu mezi klikovým čepem a vložkou.

Dokonale tuhé těleso bylo zadáno do input souboru následujícím příkazem:

```
*RIGID BODY, REF NODE=42145, ELSET=crankpin, POSITION=CENTER OF MASS
```

Kde REF NODE=42145 značí číslo referenčního uzlu, ELSET=crankpin název elementového setu, POSITION=CENTER OF MASS polohu referenčního uzlu, přemístěného do těžiště tuhého tělesa.

Nastavení okrajové symetrické podmínky na modelu ojnice

Úloha je symetrická podle roviny yz. V input souboru lze nastavit okrajovou podmínku pro rovinu symetrie příkazem:

```
*BOUNDARY  
S00004, XSYMM
```

Kde S00004 je název uzlového setu, XSYMM nastaví symetrii podle roviny y-z ($x=\text{konst.}$)

Řízení konvergence výpočtu

Při řízení konvergence výpočtu kontaktů s algoritmem Master-Slave byl do input souboru pro všechny tyto páry vložen následující příkaz:

*CONTACT CONTROLS,STABILIZE,MASTER=CM00000X, SLAVE=CS00000X

Kde CM00000X je název MASTER setu, CS00000X je název SLAVE setu.

Nastavení časového kroku pro výpočet

Při výpočtu byl změněn časový inkrement na 0.25 pro snazší ustavení kontaktů. Po ustavení kontaktů v čase 2 je časový inkrement opět nastaven defaultně na 1.0.

3. Výpočet ojnice

Výpočet napjatosti ojnice byl proveden v následujících polohách klikové hřídele:

Step 1 – montáž, Step 2 – 0°, Step 3 – 22,5°, Step 4 – 50°, Step 5 – 180°, Step 6 – 325°, Step 7 – 360°

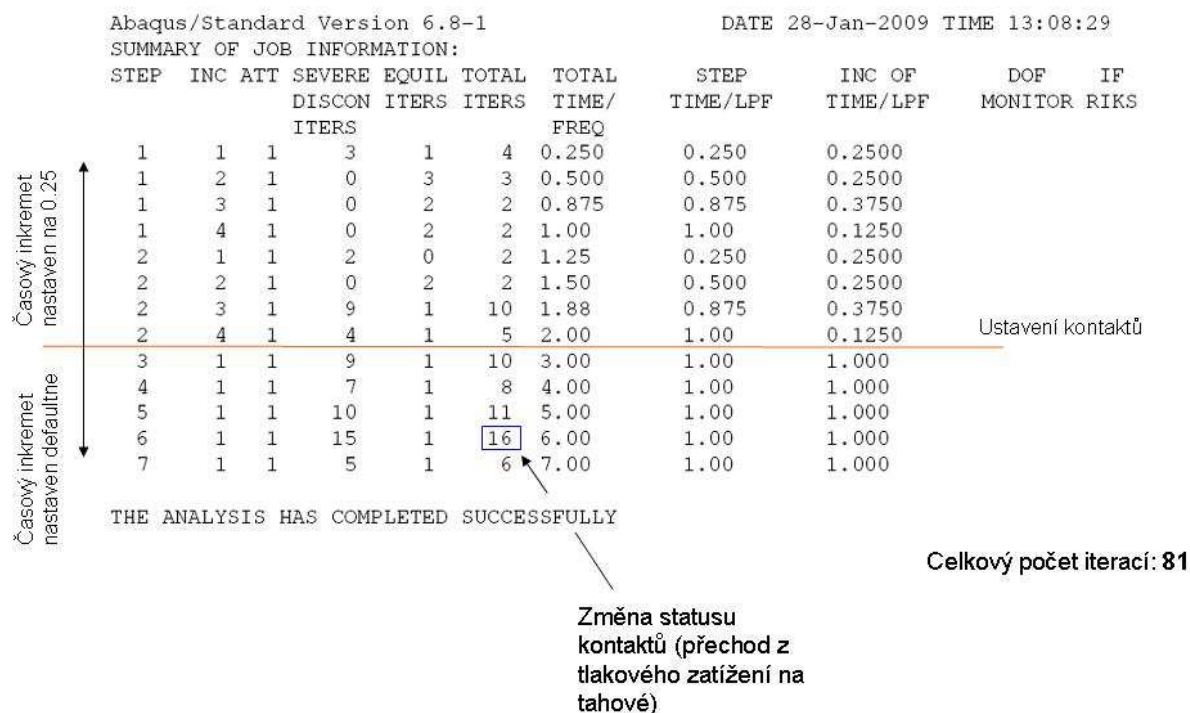
Výpočet 1

První výpočet byl proveden s nastavenou symetrickou okrajovou podmínkou a dokonale tuhým klikovým čepem. Dále byly použity výše uvedené příkazy pro kontrolu konvergence výpočtu.

Průběh výpočtu je možno vidět ze souboru, který vytvoří software Abaqus na Obr. 8. V tomto souboru jsou zapsány jednotlivé kroky výpočtu, které se dělí na inkrementy, ty jsou řešeny v několika iteracích.

U prvního výpočtu je vidět, že se v prvních dvou krocích nezačíná počítat přímo v čase 1.0, ale v 0.25. Inkrement 0.25 je použit v prvních dvou krocích záměrně, protože se v nich ustavuje kontakt. Během těchto kroků se začíná počítat s 25% celkového zatížení, což přinese snazší průběh výpočtu než kdyby se úloha začínala řešit přímo při plném zatížení.

Celkový počet iterací pro výpočet 1 je 81. Nejvíce iterací bylo potřeba provést při přechodu z pátého do šestého kroku výpočtu, kde se zatížení ojnice mění z tlakového na tahové.



Obr. 8. Výstupní soubor dokumentující průběh výpočtu 1

Výpočet 2

Druhý výpočet byl proveden se stejně upraveným input souborem jako u prvního výpočtu. Navíc byl použit příkaz pro nastavení řešiče:

*SOLUTION TECHNIQUE, TYPE=CONTACT ITERATIONS

Výstupní soubor tohoto výpočtu je zobrazen na Obr. 9. Hvězdička zobrazená u počtu iterací značí, že příkaz CONTACT ITERATIONS byl úspěšně proveden během časového inkrementu. Celkový počet iterací u výpočtu 2 je 72.

Abaqus/Standard Version 6.8-1
SUMMARY OF JOB INFORMATION:

DATE 16-Dec-2008 TIME 16:58:23

STEP	INC	ATT	SEVERE DISCON	EQUIL ITERS	TOTAL ITERS	TOTAL TIME/FREQ	STEP TIME/LPF	INC OF TIME/LPF	DOF MONITOR	IF RIKS
1	1	1		3	1	4 0.250	0.250	0.2500		
1	2	1		0	3	3 0.500	0.500	0.2500		
1	3	1		0	2	2 0.875	0.875	0.3750		
1	4	1		0	2	2 1.00	1.00	0.1250		
2	1	1U		1	0	1 1.00	0.00	1.000		
2	1	2		2	0	2 1.25	0.250	0.2500		
2	2	1		0	2	2 1.50	0.500	0.2500		
2	3	1		6*	1	7 1.88	0.875	0.3750		
2	4	1		3*	1	4 2.00	1.00	0.1250		Ustavení kontaktů
3	1	1		7*	2	9 3.00	1.00	1.000		
4	1	1		6*	1	7 4.00	1.00	1.000		
5	1	1		7*	1	8 5.00	1.00	1.000		
6	1	1		15	1	16 6.00	1.00	1.000		
7	1	1		4*	1	5 7.00	1.00	1.000		

THE ANALYSIS HAS COMPLETED SUCCESSFULLY

Celkový počet iterací: 72

Hvězdička značí, že příkaz CONTACT ITERATIONS byl úspěšně proveden během časového inkrementu

Časový inkrement nastaven na 0.25

Časový inkrement nastaven defaultně

Obr. 9. Výstupní soubor dokumentující průběh výpočtu 2

4. Závěr

Výpočet napjatosti ojnice byl proveden dvěma způsoby, při kterých bylo použito dokonale tuhého klikového čepu, okrajová symetrická podmínka a různé nastavení řešící techniky. Z hlediska počtu provedených iterací během celého výpočtu byl efektivnější výpočet s vloženým příkazem SOLUTION TECHNIQUE, TYPE=CONTACT ITERATIONS. Během tohoto výpočtu bylo ušetřeno 10 iterací oproti stejně nastavenému výpočtu bez tohoto příkazu. Obě analýzy proběhly plynule bez problémů.

Seznam použité literatury

- [1] ABAQUS Version 6.8 Documentation, Dassault Systèmes Simulia Corp., USA, 2008
- [2] Interní materiály firmy Ricardo Prague s. r. o.