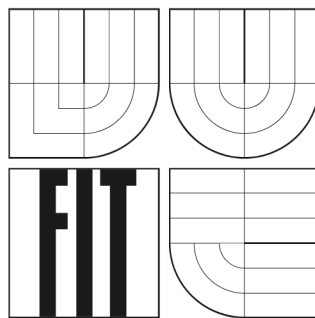


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



**Využití spektrální analýzy pro převod
trojúhelníkových polygonálních 3D sítí na 3D spline
plochy**

Diplomový projekt

2007

Miroslav Šenk

Využití spektrální analýzy pro převod trojúhelníkových polygonálních 3D sítí na 3D spline plochy

Odevzdáno na Fakultě informačních technologií Vysokého učení technického v Brně dne 24. ledna 2007

© Miroslav Šenk, 2007.

Autor práce tímto převádí svá práva na reprodukci a distribuci kopií celého díla i jeho částí na Vysoké učení technické v Brně, Fakulta informačních technologií.

Prohlášení

Prohlašuji, že jsem tuto práci vypracoval samostatně pod vedením Ing. Přemysla Krška, Ph.D. Další informace mi poskytli Ing. Michal Španěl a Ing. Ondřej Šiler. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Miroslav Šenk
24. ledna 2007

Abstrakt

Tato práce se zabývá převodem trojúhelníkových polygonálních 3D sítí na 3D spline plochy, s využitím spektrální analýzy. Zpracovávanou síť rozdělíme pomocí vlastních vektorů Laplaceova operátoru na čtyřúhelníkové oblasti. Ty budou tvořit jednotlivé spline plochy. Předvedeme některé zajímavé výsledky získané pomocí této metody a provedeme zhodnocení jejích kladů a záporů.

Klíčová slova

spline, vlastní vektory Laplaceova operátoru, Morsův-Smalův komplex

Abstract

In this work we deal with conversion of 3D triangular polygonal meshes to the 3D spline patches using spectral analysis. The converted mesh is divided into quadrilaterals using eigenvectors of Laplacian operator. These quadrilaterals will be converted into spline patches. We will present some interesting results of this method. The assets and imperfections of this method will be briefly discussed.

Keywords

spline, eigenvectors Laplacian, Morse-Smale complex

Obsah

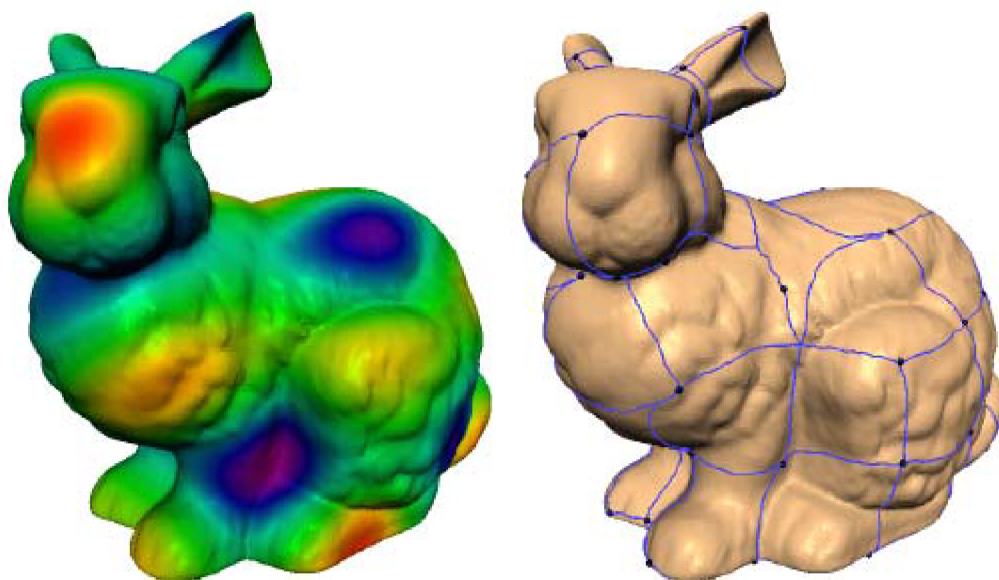
Obsah	6
1 Úvod.....	7
2 Rozbor.....	8
3 Teoretický popis metody.....	9
3.1 Spektrální analýza sítě.....	9
3.2 Vlastní čísla a vlastní vektory	11
3.3 Vytvoření čtyřúhelníkového základu	11
3.3.1 Klasifikace vrcholů	11
3.3.2 Získání Morsova-Smalova komplexu	12
3.4 Optimalizování komplexu	12
3.4.1 Topologická optimalizace	13
3.5 Quasi-Dual Komplex.....	14
3.6 Spline plochy	15
3.6.1 Spojitost spline ploch	16
3.6.2 Bézierovy plochy.....	17
3.7 Výpočet interpolačních bodů	18
4 Implementace	20
4.1 Získání vlastních vektorů Laplaceova operátoru	20
4.2 Vytvoření Morsova-Smalova komplexu a jeho optimalizace	20
4.3 Výpočet interpolačních bodů	21
5 Výsledky	22
5.1 Výsledky výpočtů vlastních čísel a vektorů.....	22
5.2 Morsův-Smalův komplex a jeho optimalizace.....	24
5.3 Quasi-Dual komplex a interpolační body.....	32
6 Závěr	43
Literatura.....	44
Příloha.....	45
Ovládání programů	45

1 Úvod

V současné době, jsou v mnoha oborech získávána trojrozměrná data, které je třeba vizualizovat. Např. ve zdravotnictví, v inženýrském modelování, laserové scany nebo satelitní data o povrchu Země.

Z prvotních získaných dat, lze například metodou marching cubes, získat trojúhelníkovou polygonální 3D síť. Polygonální síť, zvláště ty, které jsou získány skenováním, často vykazují nedostatky. Mají nedostatečné rozlišení nebo obsahují nevhodně tvarované elementy. Proto tyto síť je třeba dále zpracovat, především model vyhladit a přitom zachovat jeho podstatné vlastnosti, jako zachování tvarových detailů a objemu.

Cílem tohoto diplomového projektu bylo vyzkoušet další zpracování takového modelu převodem na model tvořený spline plochami. K tomuto účelu je vhodné nejprve převést původní trojúhelníkovou síť na síť tvořenou čtyřúhelníkovými oblastmi, které by se měly blížit čtvercům (tj. přibližně stejná délka stran a vnitřní úhly přibližně 90°). K tomuto účelu je vhodné použít novou metodu [2] využívající vlastností vlastních vektorů Laplaceova operátoru k definování skalárního pole na povrchu sítě, viz obr. 1. Toto pole je analyzováno pomocí Morsovy teorie. Pro výslednou kvalitu spline ploch je obzvláště důležité, aby rozdělení čtyřúhelníků bylo velice kvalitní. Z těchto čtyřúhelníkových oblastí získáme dalšími metodami výsledné spline plochy.



Obrázek 1: Vlastní pole Laplaceova operátoru a z něho získaný Morsuiv-Smaluiv komplex. Převzato z [2].

2 Rozbor

Další zpracování trojúhelníkových polygonálních 3D sítí může probíhat několika způsoby. Při dalším zpracování můžeme použít některou z metod dělení nebo převést model na 3D spline plochy. Základem převodu trojúhelníkových polygonálních 3D sítí na 3D spline plochy je rozdělení modelu na části, které budou reprezentovat jednotlivé spline plochy. Toho můžeme docílit několika způsoby, např. rozdělit model podle ostrých hran na povrchu, spojovat trojúhelníky do větších oblastí nebo využít spektrální analýzy.

Při rozdělování sítě na čtyřúhelníkové oblasti požadujeme, aby tato oblasti byly opravdu čtyřúhelníkové a také chceme, aby byly co nejlépe tvarovány. To samozřejmě závisí na volbě funkce $f: V \rightarrow R$, jež má přiřadit každému vrcholu sítě skalární hodnotu. Tuto funkci získáme z vlastních vektorů Laplaceovy matice L . Vlastní hodnoty $\lambda_1 = 0 \leq \lambda_2 \leq \lambda_3 \leq \dots \leq \lambda_n$ L vytváří spektrum sítě a odpovídající vlastní vektory $e_1, e_2, e_3, \dots, e_n$ L definují po částech lineární funkci přes síť M s postupně rostoucí frekvencí.

K rozdělení modelu na čtyřúhelníkový základ využijeme Morsovu teorii. Morsova teorie se zabývá analýzou sítě a pomůže nám klasifikovat vrcholy sítě na minima, maxima a sedla, pomocí předem přidělených jedinečných skalárních hodnot vrcholům. Po dopočítání hodnot vrcholů a jejich klasifikaci, můžeme vytvářet tzv. Morsův-Smalův komplex. Ten nám model rozdělí do oblastí. Jelikož Morsův-Smalův komplex bude obsahovat nedostatky, způsobené numerickým výpočtem, budeme muset provést jeho optimalizace. Z optimalizovaného Morsova-Smalova komplexu vytvoříme quasi-dual komplex, jež nám síť rozdělí na požadované čtyřúhelníkové oblasti. Tyto čtyřúhelníkové oblasti již budeme moci považovat za základ spline ploch.

V posledním kroku dopočítáme pro každou spline plochu síť interpolačních bodů, tj. bodů ležících na povrchu modelu.

3 Teoretický popis metody

3.1 Spektrální analýza sítě

Diskrétní Laplaceův operátor po částech lineární funkce přes trojúhelníkovou síť je dán vztahem:

$$\Delta f_i = \sum_{j \in N_i} w_{ij} (f_j - f_i)$$

kde N_i je množina vrcholů přiléhajících k vrcholu i a w_{ij} je skalární hodnota přidělená hraně (i, j) . Hodnota w_{ij} je dána vztahem:

$$w_{ij} = \frac{1}{2} (\cot \alpha_{ij} + \cot \beta_{ij}),$$

Kde hodnoty α_{ij} a β_{ij} jsou úhly naproti hraně (i, j) . Jestliže reprezentujeme funkci f sloupcem vektorů jeho hodnot na všech jeho vrcholech $f = [f_1 f_2 \dots f_n]^T$, můžeme přeformulovat Laplaceův operátor jako matici

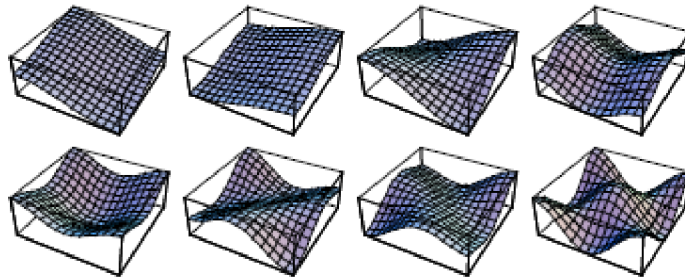
$$\Delta f = -L f$$

kde je tato matice L je definována jako

$$L_{ij} = \begin{cases} \sum_k w_{ik} & \text{jestliže } i = j, \\ -w_{ik} & \text{jestliže } (i, j) \text{ je hrana sítě } M, \\ 0 & \text{jinak.} \end{cases}$$

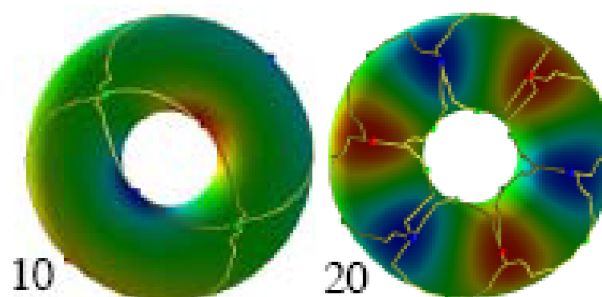
Vlastní hodnoty $\lambda_1 = 0 \leq \lambda_2 \leq \lambda_3 \leq \dots \leq \lambda_n$ L vytváří spektrum sítě a odpovídající vlastní vektory $e_1, e_2, e_3, \dots, e_n$ L definují po částech lineární funkce přes síť M s postupně rostoucí frekvencí. Tyto funkce jsou vlastní funkce Laplaceova operátoru dané sítě.

Vlastní funkce Laplaceova operátoru reprezentují přirozené harmonické kmitání tvaru, fyzicky to jsou módy kmitání povrchu. Pro rovinnou mřížku jsou vlastní vektory L základními funkcemi diskrétní cosinovy transformace (obrázek 2).



Obrázek 2: 8 prvních nekonstantních vlastních funkcí přes rovinnou mřížku 15x15 Převzato z [2].

Pro vytvoření dobře tvarovaného čtyřúhelníkového základu, mají vlastní funkce Laplaceova operátoru několik velice důležitých vlastností. Jejich kritické body jsou na povrchu modelu vhodně rovnoměrně rozmístěny, minima a maxima jsou prokládány takovými cestami, že mnohamocné vrcholy jsou velice vzácné, a vícenásobná sedla se téměř nikdy nevyskytují. Tyto vlastnosti prakticky zaručují, že mimořádné body se mohou vyskytnout pouze v extrému. Bez těchto vlastností by Morsův-Smalův komplex mohl vytvořit velice nekvalitní čtyřúhelníkový základ. Další důležitou vlastností vlastních funkcí je také to, že se vyskytují v řádu s rostoucím kmitočtem a tudíž s rostoucím počtem kritických bodů. Z toho plyne, že počet uzlových domén vlastní funkce s vlastní hodnotou λ_k je nejvíce k . Díky této vlastnosti je relativně jednoduché určit, který vlastní vektor zvolit pro určení funkce f . Proto nám také stačí spočítat prvních 40 – 160 vlastních vektorů (obr. 3). Podle toho, jak je zpracováváný model velký a jak tvarovaný. Tohoto se dá také využít při výpočtu vlastních čísel a vlastních vektorů, který je velice časově náročný, a při volbě algoritmu vybrat takový, který nám opravdu umožní spočítat pouze prvních k vlastních hodnot.



Obrázek 3: Ukazuje prstenec a jeho vlastní pole, pro jeho 10. a 20. vlastní vektor. Převzato z [2].

3.2 Vlastní čísla a vlastní vektory

Komplexní číslo λ nazýváme vlastním číslem čtvercové matice A typu $(n \times n)$, existuje-li nenulový vektor X takový, že

$$A \cdot X = \lambda \cdot X.$$

Takový vektor X se nazývá vlastním vektorem matice A odpovídajícím vlastnímu číslu λ . Vlastní vektor není určen jednoznačně, vždy je jich nekonečně mnoho. Vlastní čísla matice A lze získat řešením tzv. charakteristické rovnice matice A :

$$\det(A - \lambda E) = 0.$$

Vlastní vektory pak lze určit řešením homogenní soustavy lineárních algebraických rovnic

$$(A - \lambda E) \cdot X = 0.$$

Matice A má vlastní číslo 0, právě když je singulární. Naopak, nula není vlastním číslem matice A , právě když A je regulární. Existuje-li inverzní matice A^{-1} , je λ vlastním číslem matice A právě když $1/\lambda$ je vlastním číslem matice A^{-1} . Vlastní vektory matice A odpovídající vlastnímu číslu λ a vlastní vektory matice A^{-1} odpovídající vlastnímu číslu $1/\lambda$ jsou stejné. Je-li λ vlastním číslem matice A a X je příslušný vlastní vektor, pak λ^2 je vlastním číslem matice A^2 a X je opět příslušným vlastním vektorem. Nenulový vektor X (jehož všechny složky jsou reálná čísla) je vlastním vektorem matice A právě když při lineárním zobrazení \mathcal{L} reprezentovaném maticí A jsou vektory X a $\mathcal{L}(X)$ ve $V(E_n)$ lineárně závislé (tj. jsou kolineární).

Necht' A je čtvercová matice typu $(n \times n)$. Množinu $\{\lambda_1, \lambda_2, \dots, \lambda_n\}$ všech vlastních čísel matice A nazýváme spektrem matice A a značíme $\sigma(A)$. Číslo $\rho(A) = \max\{|\lambda_1|, |\lambda_2|, \dots, |\lambda_n|\}$ nazýváme spektrálním poloměrem matice A .

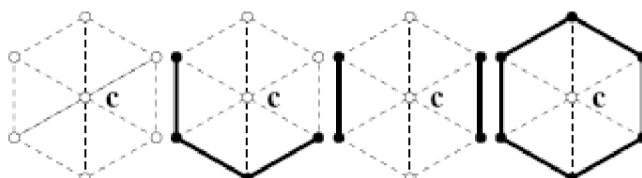
Všechna vlastní čísla symetrické matice jsou reálná.

3.3 Vytvoření čtyřúhelníkového základu

3.3.1 Klasifikace vrcholů

Klasifikace vrcholu probíhá na základě jeho hodnoty a hodnoty jeho okolí. Okolí tvoří vrcholy, které jsou s daným vrcholem spojeny hranou. Vrchol je klasifikován jako minimum, je-li jeho hodnota menší než hodnota všech okolních vrcholů. Má-li vrchol hodnotu větší než všechny jeho okolní vrcholy, jedná se o maximum. Dojde-li ke čtyřem

změněm mezi větší a menší hodnotou vrcholu a jeho okolí, jde o sedlo. V ostatních případech je vrchol klasifikován jako regulární (obr. 4).



Obrázek 4: Klasifikace vrcholu c . zleva: minimum, regulární bod, sedlo a maximum. Plný bod značí vyšší hodnotu než c , prázdný bod hodnotu nižší. Převzato z [2].

3.3.2 Získání Morsova-Smalova komplexu

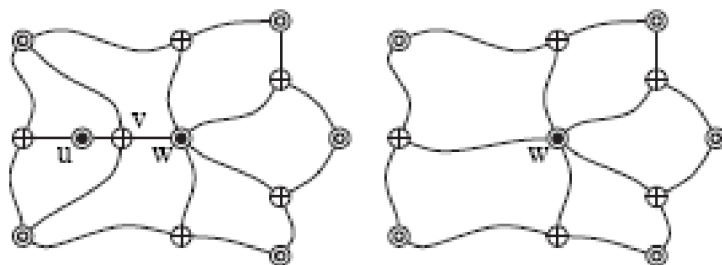
Po dopočítání hodnot vrcholů a jejich klasifikaci, můžeme vytvářet tzv. Morsův-Smalův komplex. Ten nám model rozdělí do oblastí. Morsův-Smalův komplex vytváříme tak, že pro každé sedlo dopočítáme čtyři nejstrmější cesty, které dané sedlo spojují s okolními extrémami. Dvě cesty jsou stoupající a spojují sedlo s maximy a dvě jsou klesající a spojují sedlo s minimy. Po dopočítání všech cest by měla být síť rozdělena na buňky. Každá buňka by měla obsahovat právě dvě sedla, jedno maximum a jedno minimum.

3.4 Optimalizování komplexu

Po získání Morsova-Smalova komplexu výše popsaným postupem zjistíme, že komplex není zcela vhodný a že ho je třeba optimalizovat. Toto je způsobeno numerickým výpočtem vlastních vektorů. Ten zanechá do vlastního pole nějaký šum, který způsobí, že vzniknou nežádoucí kritické body. Tyto chyby odstraňujeme použitím dvou optimalizačních procesů. Nejprve optimalizujeme topologii komplexu odstraněním dvoumocných maxim a minim, posléze odstraňujeme dvojice sedla maximum či sedlo minimum, které jsou na modelu příliš blízko sebe.

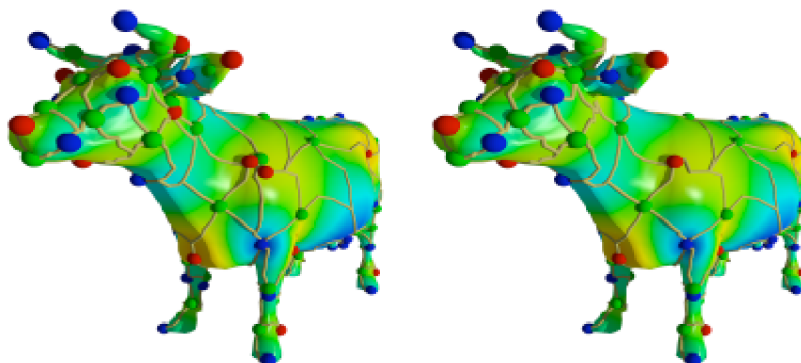
3.4.1 Topologická optimalizace

K odstranění nepatřičných kritických bodů používáme *rušení*. Při každém rušení je odstraněna propojená dvojice sedlo a minimum nebo maximum. V prvním kroku rušíme dvojmocná maxima nebo minima a sedlo s ním spojené. Sedlo vybereme na základě *trvalosti*. Trvalost je určena jako absolutní hodnota rozdílu funkčních hodnot dvojice sedlo a minimum nebo maximum. Vždy volíme to sedlo, které má s rušeným maximumem nebo minimumem menší trvalost. Při této operaci zrušíme také dvě cesty vedoucí ze sedla do stávajících dvou maxim či dvou minim. Příklad takového rušení můžeme vidět na obr. 5.



Obrázek 5: Na obrázku můžeme vidět dvojmocné maximum u , které spolu se sedlem v a dvěma cestami do přilehlých minim zrušíme. Převzato z [2].

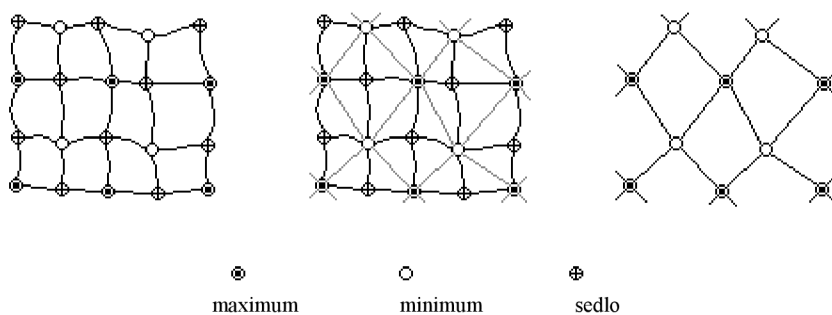
Další rušení můžeme provádět na základě trvalosti nebo na základě geometrické *vzdálenosti*. Rušení na základě trvalosti je popsáno v Dong et al. [2] a probíhá tak, že se dvojice s nejnižší trvalostí je zrušeno nejdříve. Jak si ale později ukážeme, při tomto rušení dochází k odstranění potřebných kritických bodů (obr. 22). Z tohoto důvodu jsem navrhl rušení na základě vzdálenosti. Vzdálenost je definována jako geometrická vzdálenost dvojice sedlo a maximum či minimum. Dvojice s nejmenší vzdáleností jsou zrušeny nejdříve. Ukázkou takového rušení můžete vidět na obr. 6.



Obrázek 6: Na modelu krávy (vlevo) lze vidět nežádoucí kritický bod nad levou přední nohou. Model vpravo je po topologické optimalizaci na základě vzdálenosti. Převzato z [2].

3.5 Quasi-Dual Komplex

Z Morsova-Smalova komplexu vytvoříme quasi-dual komplex. Jelikož víme, že všechny sedla jsou čtyřmocná, můžeme vytvořit buňky ze dvou maxim a dvou minim přilehlých ke každému sedlu. Takto vzniklý komplex bude obsahovat asi jen polovinu vrcholů, pouze minima a maxima, a počet buněk bude odpovídat počtu sedel. Ukázku můžeme vidět na obr. 7. Quasi-dual komplex je více efektivnější a kompaktnější pro další zpracování.



Obrázek 7: Vytvoření quasi-dual komplexu z Morsova-Smalova komplexu.

3.6 Spline plochy

Bodovou funkci parametrické plochy $Q(u, v)$ s parametry u a v lze zapsat jako:

$$Q_{3D}(u, v) = [x(u, v), y(u, v), z(u, v)]$$

Kde $x(u, v)$, $y(u, v)$ a $z(u, v)$ jsou funkce dvou parametrů u a v . Tyto parametry mohou nabývat hodnot z rozsahu 0 až 1. Jinými slovy znamená zápis bodové funkce fakt, že bodu o souřadnicích $[x, y, z]$ v trojrozměrném prostoru odpovídá bod o souřadnicích $[u, v]$ v prostoru parametrickém.

Na parametrickou plochu se můžeme dívat jako na množinu bodů vzniklou tažením křivky po určité trajektorii. Tato křivka při svém pohybu může měnit tvar. Z bodové rovnice lze jednoduše vyjádřit rovnice tečných vektorů ve směrech parametrů u a v k ploše $Q(u, v)$ a z těchto dvou vektorů vypočítat normálu k povrchu:

$$n = \frac{(q_u \times q_v)}{|q_u \times q_v|}$$

Kde \times znamená operaci vektorového součinu.

Napojování parametrických ploch

Pro účely počítačové grafiky i dalších (například fyzikálních či estetických) požadavků je důležité zaručit spojitost napojení dvojice plátů. Rozlišujeme dva druhy spojitosti - *parametrickou spojitost* C^m a *geometrickou spojitost* G^m .

Dva pláty parametrické plochy mají napojení C^0 , mají-li společnou hranu, která je křivkou třídy alespoň C^0 . Dva pláty mají spojitě napojení C^1 , pokud mají společnou jednu stranu a mají-li shodné parciální derivace ve všech bodech společné strany prvního i druhého plátu.

Dva pláty mají spojitě napojení G^1 , mají-li společnou hranu, která je křivkou spojitosti alespoň G^1 (pro zaručení geometrické spojitosti G^1 je zapotřebí zachovat alespoň spojitou změnu tečen, nikoli tečných vektorů, jak je tomu u spojitosti C^1) a jsou-li parciální derivace podél této strany ve směru napojení lineárně závislé s koeficientem $k > 0$, který se spojitě mění podél této společné strany.

Pro většinu aplikací je pro hladké napojení plátů nutné dodržet alespoň spojitost G^1 , pouze pro některé specializovanější aplikace se musí dodržet přísnější podmínky spojitosti C^1 .

Modelování parametrických ploch

Parametrické plochy se při interaktivním modelování zadávají pomocí *řídících bodů* a *bázových funkcí*. Bázové funkce jsou většinou přímo dány použitým typem křivek a ploch, uživatel tedy může měnit pouze polohu řídících bodů.

V počítačové grafice se ponejvíce používají *aproximační plochy*. *Interpolační plochy* se pro modelování v trojrozměrném prostoru většinou nepoužívají, neboť mají pro většinu úloh počítačové grafiky nevhodné vlastnosti - nežádoucí oscilace, nelokálnost změn při posunu řídících bodů atd.

Aproximace a interpolace s použitím parametrických ploch

Při aproximaci určuje poloha řídicích bodů určitým způsobem tvar výsledné plochy, i když tato plocha obecně těmito body nemusí procházet. Pro napojování ploch je důležité, aby byly specifikovány tečné podmínky na všech jejich stranách. Jako bázové funkce se nejčastěji používají polynomy, které však nutně nemusí mít ve směrech růstu parametrů u a v stejný stupeň.

Mezi často požadované vlastnosti parametrických křivek a ploch patří:

Invariance k lineárním transformacím popř. i perspektivní projekci. Tato vlastnost zaručuje, že lineární transformace popř. projekce aplikované na řídicí body parametrické plochy má stejný výsledek, jako aplikace této transformace na každý bod vygenerované plochy. Je zřejmé, že při splnění podmínky invariance je z výkonnostního hlediska výhodnější aplikovat transformace pouze na řídicí body parametrické plochy. Většina používaných aproximačních ploch podmínku invariance splňuje vzhledem k lineárním transformacím, u racionálních ploch (mezi něž patří i NURBS (*Non-Uniform Rational B-Splines*) plochy) je současně splněna i invariance vzhledem k perspektivní projekci. Bézierovy plochy (neracionální) však nejsou invariantní vzhledem k perspektivní projekci.

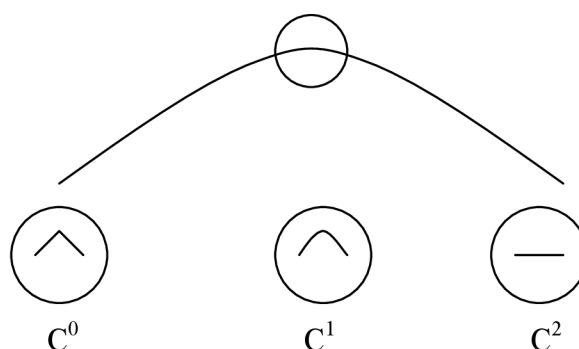
Vlastnost konvexní obálky zaručuje, že všechny body vygenerované parametrické křivky či plochy leží v konvexní obálce všech svých řídicích bodů. Může se použít i slabší podmínka, kde pouze část plochy leží v konvexní obálce některých řídicích bodů. Pokud zaručíme tuto vlastnost, zjednoduší se velké množství algoritmů prováděných s modely těles, například test na průnik dvou těles, vytváření obalových těles nebo test průsečíku paprsku s tělesem. Bézierovy plochy leží uvnitř konvexní obálky. Totéž platí pro racionální Bézierovy plochy, ovšem za předpokladu, že jsou váhy všech řídicích bodů kladné.

Plocha může procházet **krajními body** svého řídicího polygonu. Tuto vlastnost, která je zaručena například u Bézierových ploch, lze využít pro snadné napojování jednotlivých plátů s dodržением požadované třídy spojitosti C^0 , C^1 , G^1 nebo G^2 . Této vlastnosti lze dosáhnout i u B-spline ploch použitím takzvaných násobných řídicích bodů, tj. bodů majících stejné souřadnice v prostoru.

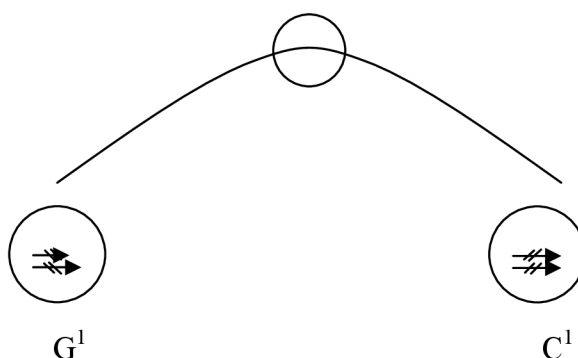
V současnosti se v počítačové grafice používá několik navzájem odlišných typů parametrických ploch, které se liší jak svými geometrickými vlastnostmi, tak i implementační, výpočetní a paměťovou náročností.

3.6.1 Spojitost spline ploch

Při napojování spline ploch je velice významným faktorem spojitost. Při napojování plátů se může podle parametrů aplikace požadovat různý stupeň spojitosti. Pokud mají dva pláty společnou alespoň jednu hranu, mají napojení typu C^0 . Pokud mají dva pláty společnou jednu hranu a současně jsou shodné i parciální derivace ve všech bodech společné hrany, jedná se o napojení typu C^1 , obrázek 8. Někdy je dostačující napojení typu G^1 , u kterého je zapotřebí zachovat spojitou změnu tečen - tečné vektory musí být kolineární, obrázek 9.



Obrázek 8: Spojitosti C^n



Obrázek 9: Spojitosti G^1 a C^1

3.6.2 Bézierovy plochy

Bézierova plocha nm stupně je určena $(n+1) \times (m+1)$ řídicími body $P_{i,j}$ a vztahem:

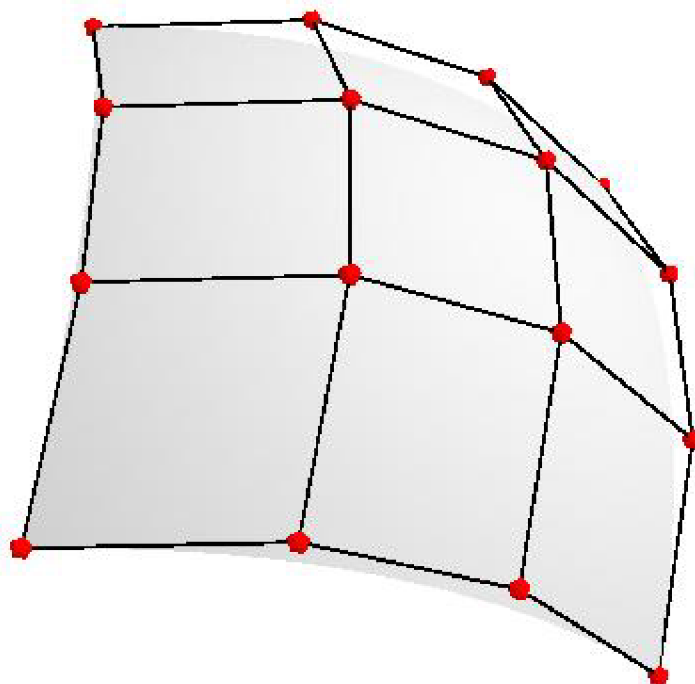
$$Q(u, v) = \sum_{i=0}^n \sum_{j=0}^m P_{i,j} B_i^n(u) B_j^m(v)$$

Kde indexy i resp. j probíhají přes intervaly $i=0, \dots, n$ a $j=0, \dots, m$. Bázové funkce $B_i^n(u)$ a $B_j^m(v)$ jsou představovány Bernsteinovými polynomy n resp. m stupně.

Nejpoužívanějším typem těchto ploch jsou v návrhových aplikacích Bézierovy bikubické pláty, u kterých jsou použity Bernsteinovy polynomy třetího stupně. Tyto parametrické plochy jsou zadány pomocí šestnácti řídicích bodů, které tvoří mřížku o velikosti 4×4 body. Bernsteinovy polynomy vyšších stupňů lze sice také použít (počet řídicích bodů roste kvadraticky se stupněm polynomu), ale zvyšuje se výpočetní složitost.

U Bézierových bikubických plátů je možné při jejich navazování využít jejich specifických vlastností. Okraje bikubických plátů jsou tvořeny Bézierovými křivkami, jejichž

řídící body jsou určeny čtveřicí řídících bodů bikubického plátu. Na samotném plátu také existuje několik dalších Bézierových křivek. Ty jsou určeny vždy čtveřicí řídících bodů, které leží v jedné řadě či v jednom sloupci. Bézierův bikubický plát prochází svými čtyřmi rohovými body, což je vidět na obrázku 10. Pokud je zaručeno, že dva pláty mají společnou hranu, tj. jejich čtyři krajní body jsou totožné, má výsledná plocha spojitost C^0 - mezi plochami nejsou mezery ani přesahy. Pokud je zaručena i kolinearita tečných vektorů na okraji ploch, má výsledná plocha spojitost G^1 .



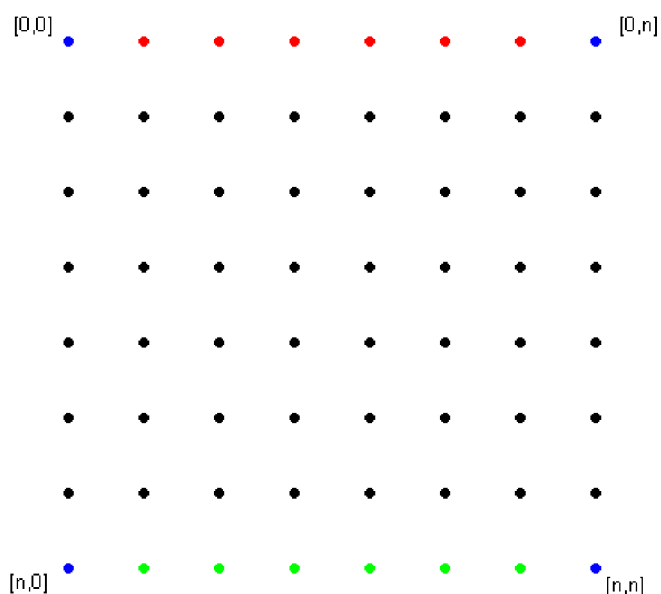
Obrázek 10: Bézierův bikubický plát specifikovaný 16 řídícími body

3.7 Výpočet interpolačních bodů

Jelikož quasi-dual komplex vytvoří na modelu kvalitní čtyřúhelníkový základ, můžeme tento základ považovat za základ splínů a začít počítat interpolační body. To jsou body ležící na povrchu modelu.

Pro každý spline spočítáme $n \times n$ interpolačních bodů. Rohové body známe, dále postupně dopočítáme body $[0,1]$ až $[0,n-1]$, které tvoří jednu stranu spline plochy (na obr. 11 označeny červeně). V dalším kroku body $[n,1]$ až $[n,n-1]$, které tvoří protější stranu, naproti

první dopočítané, spline plochy (na obr. 11 označeny modře). A nakonec postupně $[0,0]$ až $[n,0]$, $[0,1]$ až $[n,1]$, \dots , $[0,n]$ až $[n,n]$.

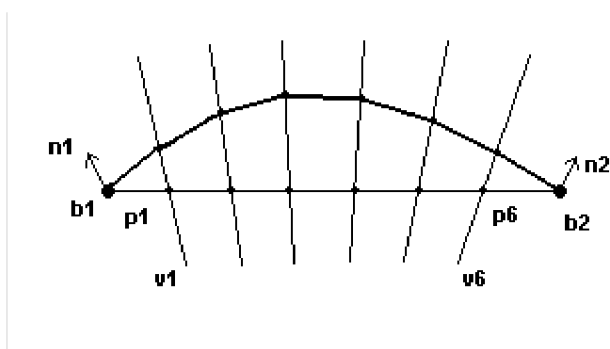


Obrázek 11: Matice interpolačních bodů pro $n=8$.

A to následovně: spočítáme průsečíky přímek definované bodem p_n a vektorem v_n s povrchem modelu (obrázek 12). p_n a v_n jsou definovány vztahem

$$p_n = b_1 + \left(\frac{b_2 - b_1}{n-1} \right) * k, \text{ kde } k = 1, 2, \dots, 6 \text{ pro } n = 8, b_1 \text{ a } b_2 \text{ jsou krajní body,}$$

$$v_n = n_1 + \left(\frac{n_2 - n_1}{n-1} \right) * k, \text{ kde } k = 1, 2, \dots, 6 \text{ pro } n = 8, n_1 \text{ a } n_2 \text{ jsou normály krajních bodů.}$$



Obrázek 12: Výpočet interpolačních bodů pro $n=8$.

4 Implementace

Danou metodu jsem implementoval strukturálně v C++. Zpracování dat jsem prováděl pomocí dvou aplikací. První aplikace, počítá vlastní čísla a vlastní vektory a druhá zobrazuje získaná data, vytvoří Smalův-Morsův komplex, optimalizuje ho, vytvoří quasi-dual komplex a dopočítá interpolační body. Zpracovávaná data jsem získal z modelů, které mi poskytl dr. Kršek. Ty jsou uloženy ve formátu definovaném knihovnou VectEntity2, jejíž autorem je dr. Kršek.

4.1 Získání vlastních vektorů Laplaceova operátoru

Jelikož knihovna VectEntity2 uchovává pouze vrcholy a trojúhelníky, jež jsou uloženy v lineárních seznamech, uchovávat hrany není nutné, prvním mým krokem bylo vytvoření hran (funkce VytvoreniHran). Ty jsou vhodné pro dočasné přechování hodnoty w_{ij} (funkce OhodnoceniHran), která je použita pro následné vytvoření matice L . Tu je nejvhodnější vytvořit dvojitým průchodem sítě po vrcholech (funkce VytvorMatici).

Jedním z nejtěžších kroků je získání samotných vlastních čísel a vlastních vektorů této matice. Za tímto účelem jsem implementoval několik metod. První metoda počítá vlastní čísla a vlastní metody pomocí Jacobiho rotací. Tato metoda je sice spolehlivá, ale je velice pomalá a pro rozsáhlé matice je takřka nepoužitelná. Druhá metoda, kterou jsem implementoval a nakonec i použil, je QL algoritmus s implicitními posuny [4], který počítá vlastní čísla a vlastní vektory tridiagonální matice (funkce tqli). Tu získám z matice L Householderovou redukcí [4] (funkce tred2). Všechny tyto výpočty jsou prováděny v aplikaci *vypocet*. Dále jsem se, na doporučení dr. Krška, pokoušel využít spojení knihoven Atlas a lapack, které mi poskytl Ing. Španěl. Ale jak si ukážeme v kapitole výsledky, hodnoty spočtené touto knihovnou byly nevhodné.

4.2 Vytvoření Morsova-Smalova komplexu a jeho optimalizace

Klasifikaci vrcholů provádím ve funkci KlasifikaceVrcholu. Při klasifikaci srovnávám hodnotu každého vrcholu s hodnotami vrcholů v jeho okolí. Tyto vrcholy mám seřazeny proti směru hodinových ručiček, jak jsou okolo aktuálního vrcholu.

Morsův-Smalův komplex je dopočítáván tak, že z každého sedla najdou čtyři nejstrmější cesty. Střídavě dvě klesající, vedoucí k minimům, a dvě stoupající, vedoucí k maximům.

Optimalizace jsem řešil podle dříve uvedeného postupu. Ve funkci TopOp provádím topologickou optimalizaci. Nejdříve každé dvojici sedlo-minimum nebo sedlo-maximum přiřadím absolutní hodnotu rozdílu jejich funkčních hodnot a posléze tyto hodnoty normalizuji na hodnotu 0 až 100. Posléze vyhledám všechny dvojmocná minima a maxima. Spolu s dvojmocným minimem či maximem odstraním to sedlo, které má nižší trvalost, a to podle výše popsaného algoritmu. V posledním kroku optimalizací každé dvojici sedlo extrém určím geometrickou vzdálenost. Nakonec odstraním ty dvojice, které mají menší hodnotu než je hodnota zadaná uživatelem.

4.3 Výpočet interpolačních bodů

Výpočet interpolačních bodů jsem prováděl podle algoritmu popsaného v kapitole 3.7. Při hledání průsečíku přímky s povrchem modelu jsem postupoval tak, že jsem vždy našel průsečík přímky s rovinou jednoho trojúhelníku modelu, a pak jsem zjišťoval, zda tento průsečík leží v tomto trojúhelníku, když ne, tak jsem použil sousední trojúhelník. Jako první trojúhelník jsem použil ten, který byl přilehlý k výchozímu bodu. Při tomto výpočtu jsem nepočítal pouze interpolační body, ale použil jsem i tzv. zjemnění, to znamená, že jsem dopočítal také několik mezibodů mezi jednotlivými interpolačními body, které mi pomáhaly vybírat správnou cestu po trojúhelnících modelu při hledání průsečíku s povrchem modelu.

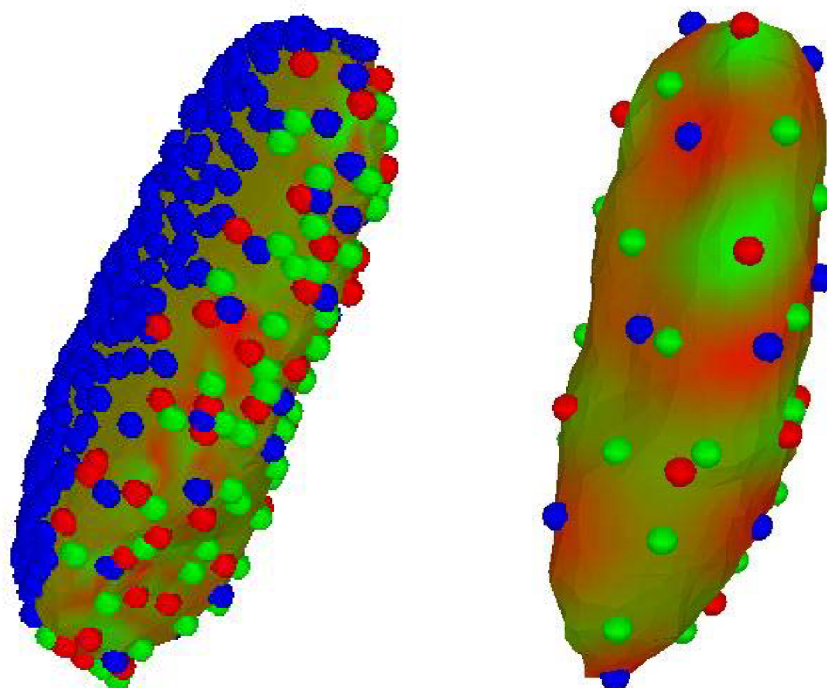
5 Výsledky

5.1 Výsledky výpočtů vlastních čísel a vektorů

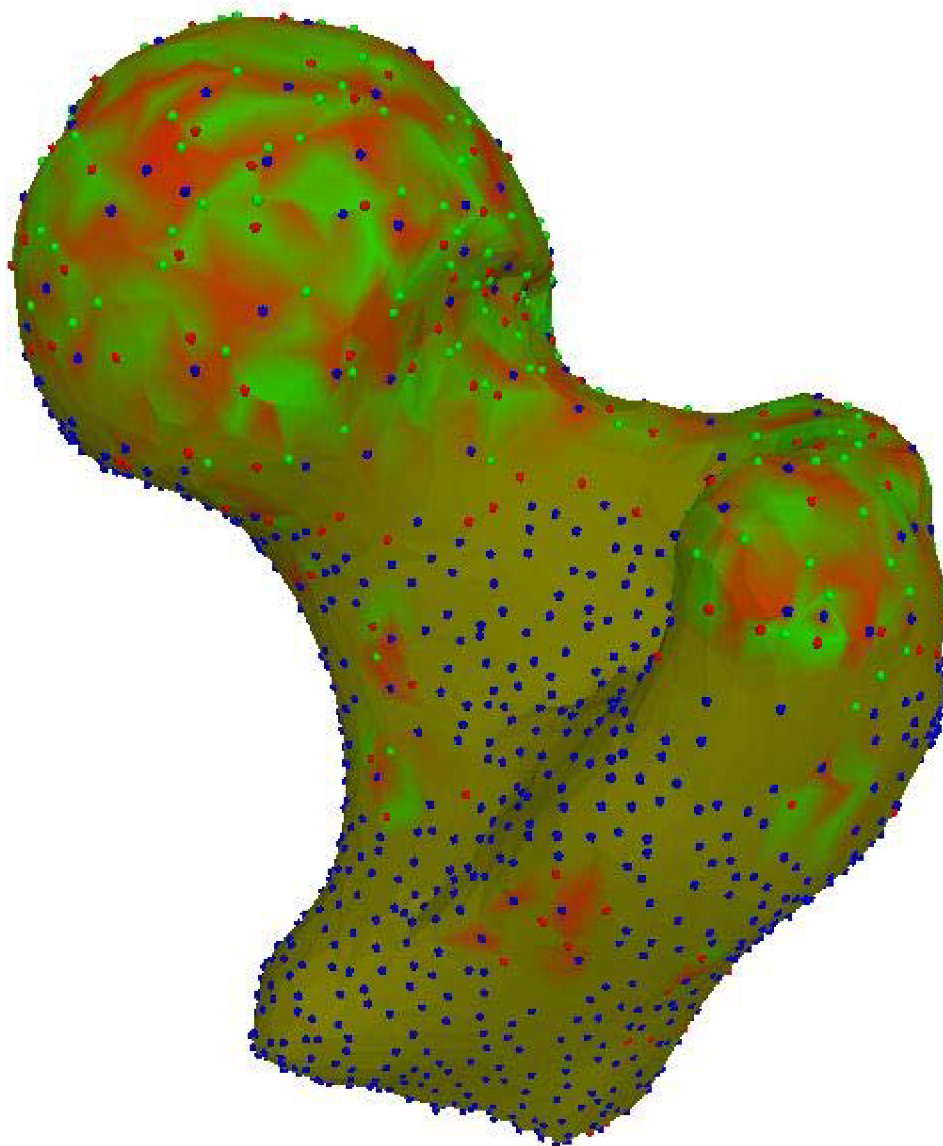
Následující tabulka 1 obsahuje srovnání přibližných časů výpočtů vlastních čísel a vlastních vektorů různých modelů. Jak je v tabulce vidět, QL algoritmus je ve srovnání s knihovnou lapack velice pomalý. Avšak je nutné si uvědomit, že výsledky získané knihovnou lapack jsou nevhodné, a proto jsem s těmito výsledky dále nepracoval. Srovnání výsledků spektrální analýzy a rozmístění kritických bodů těchto dvou metod výpočtů můžeme vidět na obrázcích 13 až 16.

počet vrcholů modelu	1000	2500	5000
QL algoritmus	6s	140s	12min
knihovna lapack	105s	45min	více než 13hod.

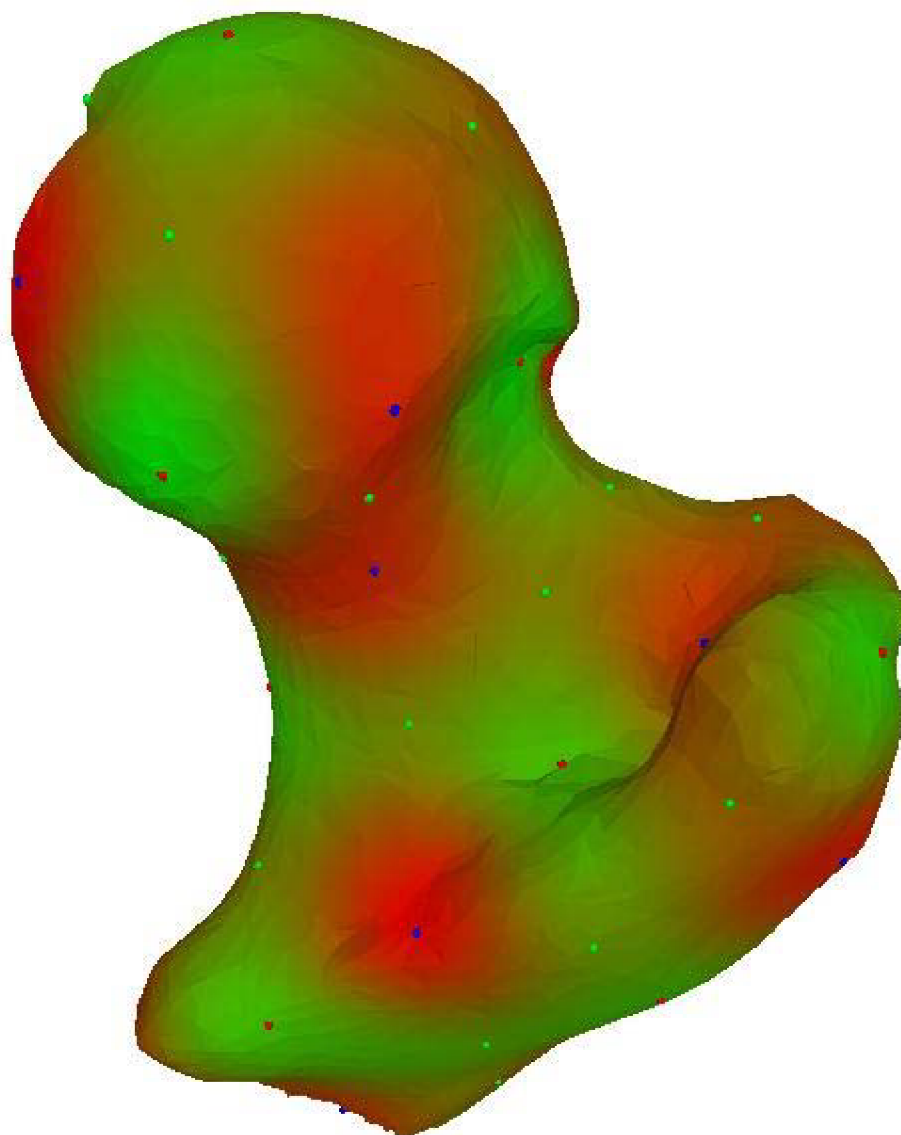
Tabulka 1: Přibližné časy výpočtů vlastních čísel a vektorů.



Obrázky 13 a 14: Rozmístění kritických bodů na modelu zubu (sedla jsou zelené, minima červeně a maxima modře). Vlevo použit pro výpočet 50. vlastního vektoru lapack , vpravo QL algoritmus.



Obrázek 15: Model kloubu, použit 50. vlastní vektor vypočítaný knihovnou lapack.

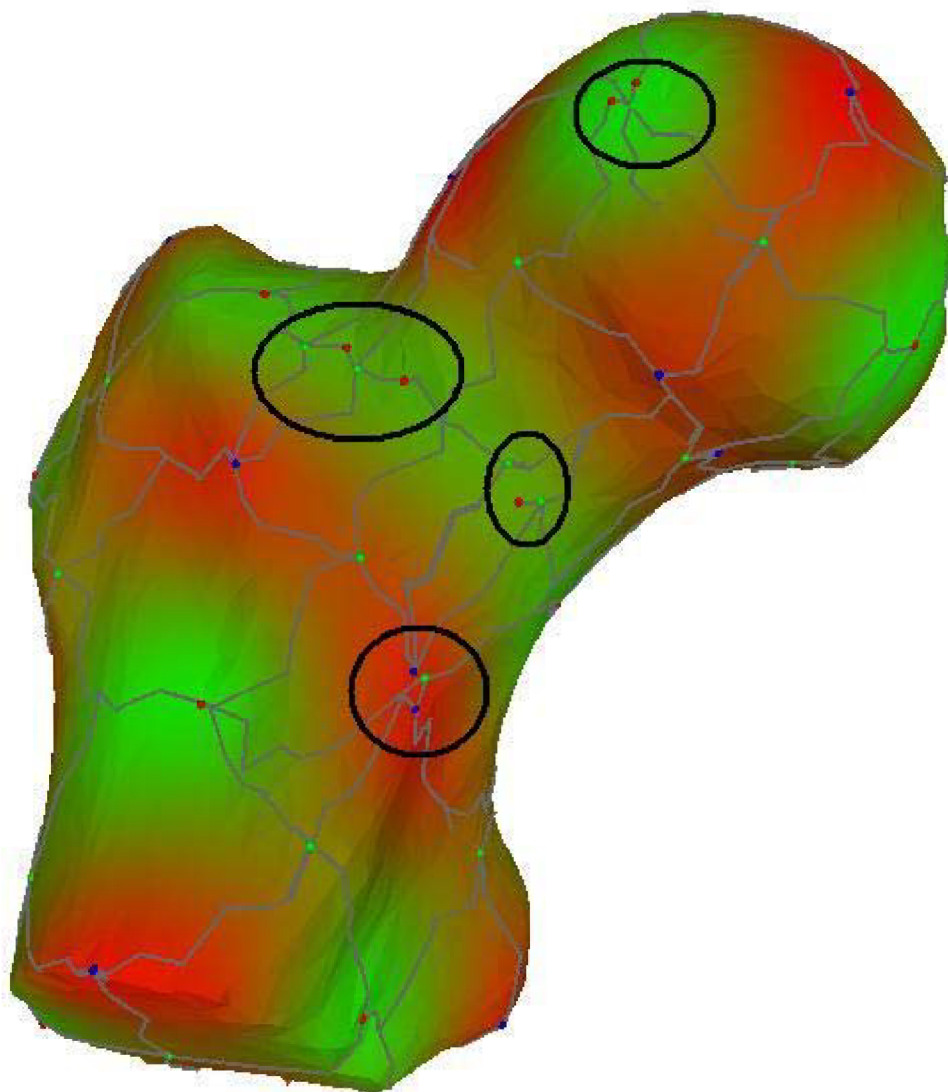


Obrázek 16: Model kloubu, použit 50. vlastní vektor vypočítaný QL algoritmem.

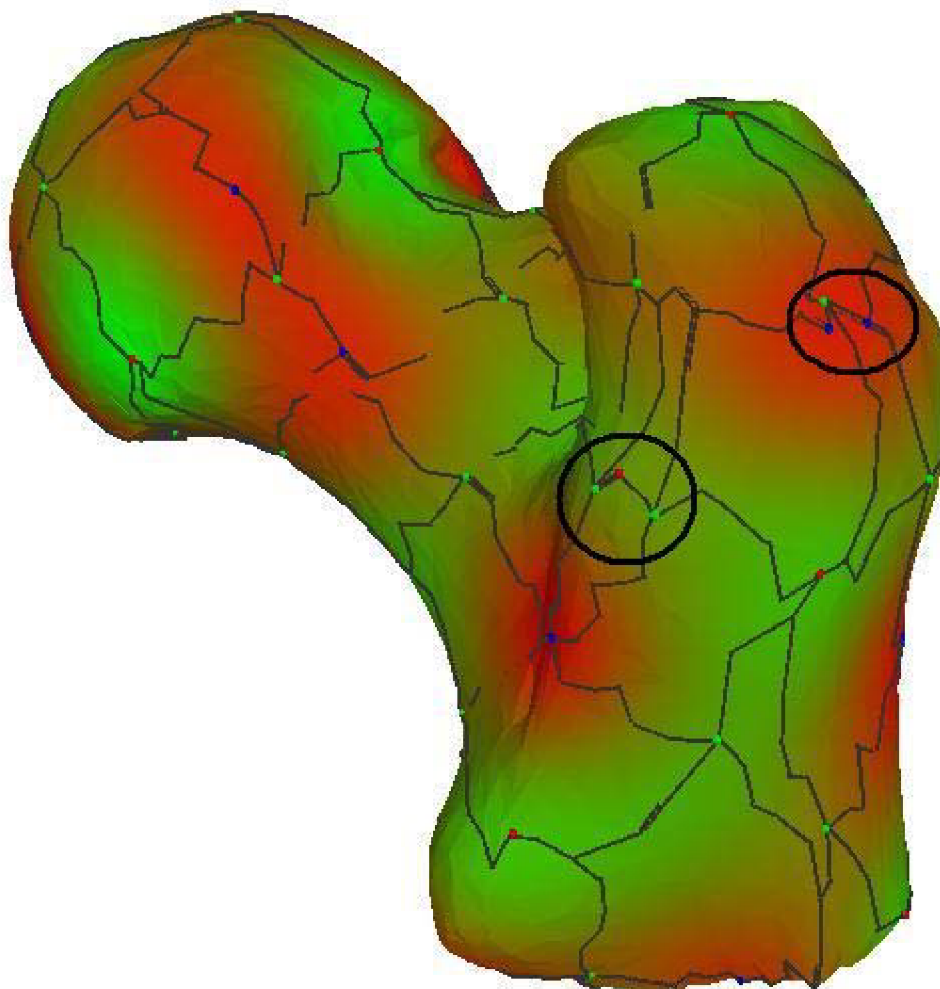
5.2 Morsův-Smalův komplex a jeho optimalizace

V této části si ukážeme Morsův-Smalův komplex a postupné výsledky jeho optimalizace. Na obrázku 17 a 18 je vidět Morsův-Smalův komplex bez optimalizací na modelu kloubu. Nevhodné shluky kritických bodů jsou zakroužkovány. Na obrázku 19 jaké nevhodné čtyřúhelníky z takového komplexu vzniknou. Po odstranění dvoumocných minim a maxim

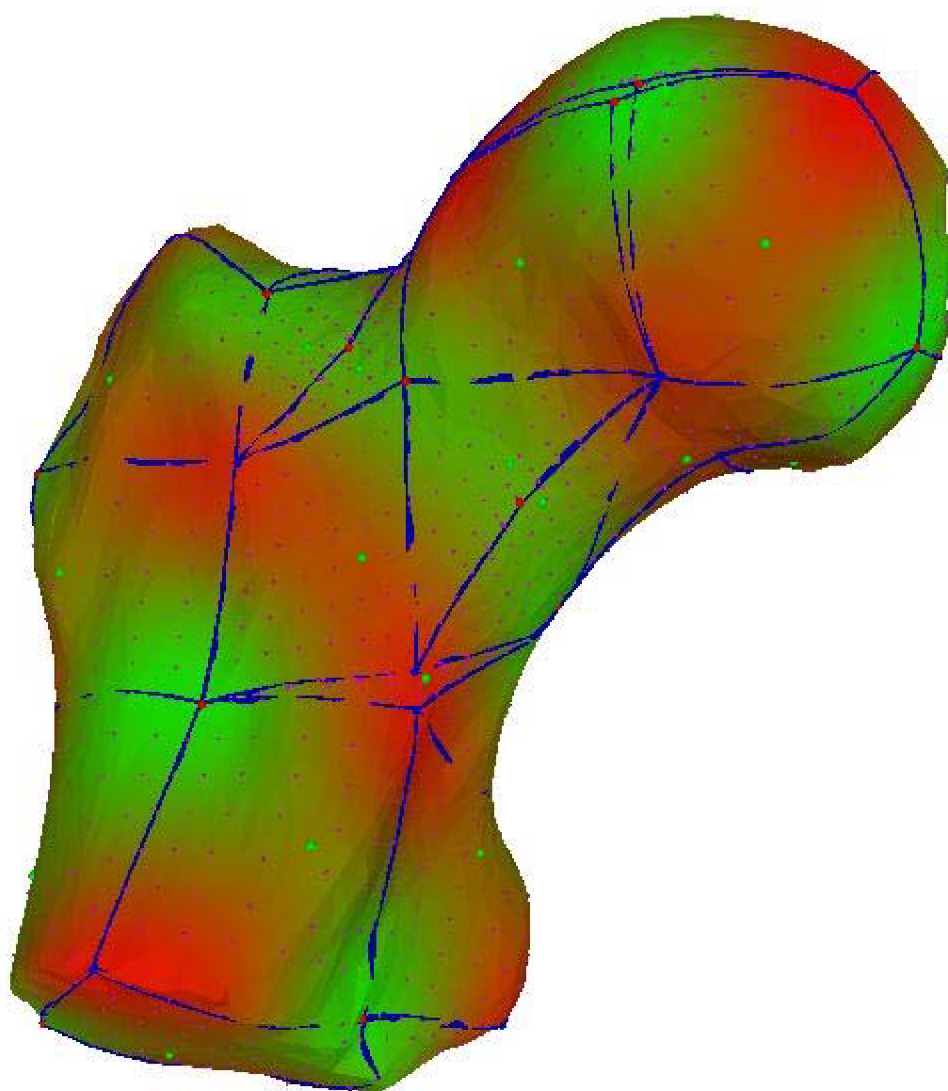
můžeme pozorovat znatelné zlepšení, ale komplex stále vykazuje nedostatky (obr. 20 a obr.21). Na obrázku 22 je vidět nevhodnost optimalizace na základě trvalosti, ačkoli bylo zrušeno potřebné sedlo, komplex stále vykazuje nedostatky. Tento problém odstraňuje mnou navržené rušení na základě geometrické vzdálenosti (obr. 23).



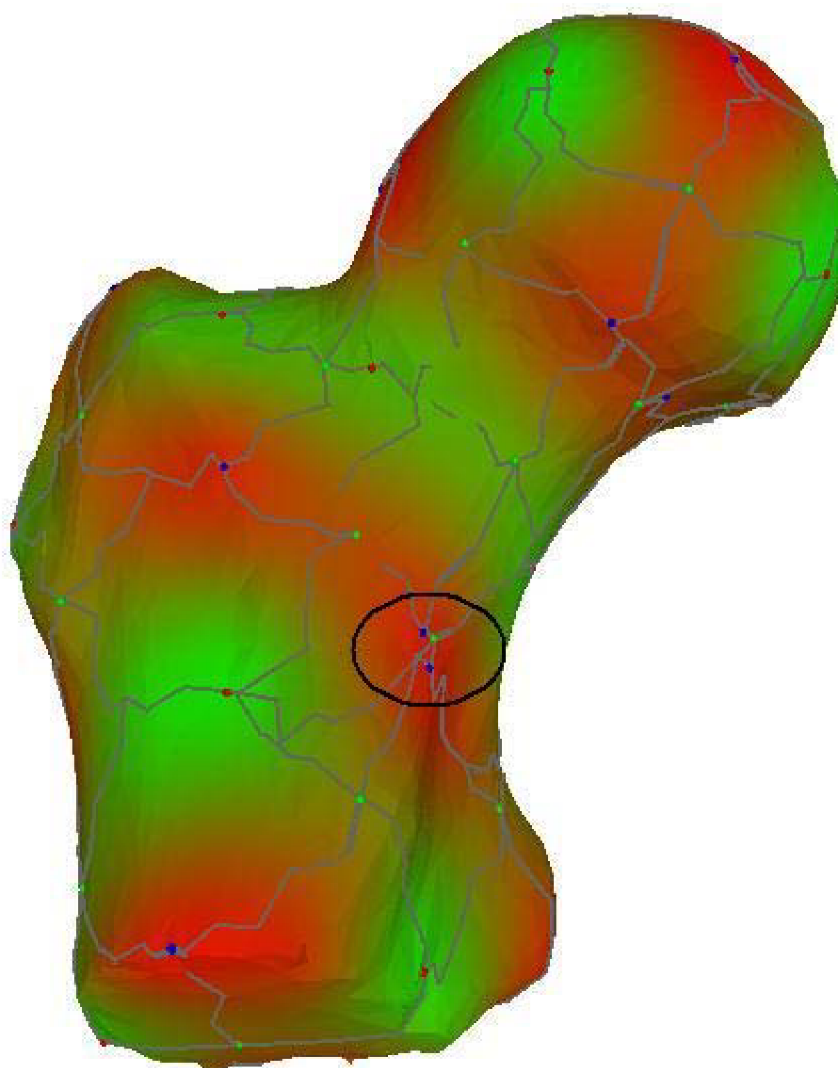
Obrázek 17: Model kloubu a jeho Morsův-Smalův komplex bez optimalizací. Nedostatky zvýrazněny.



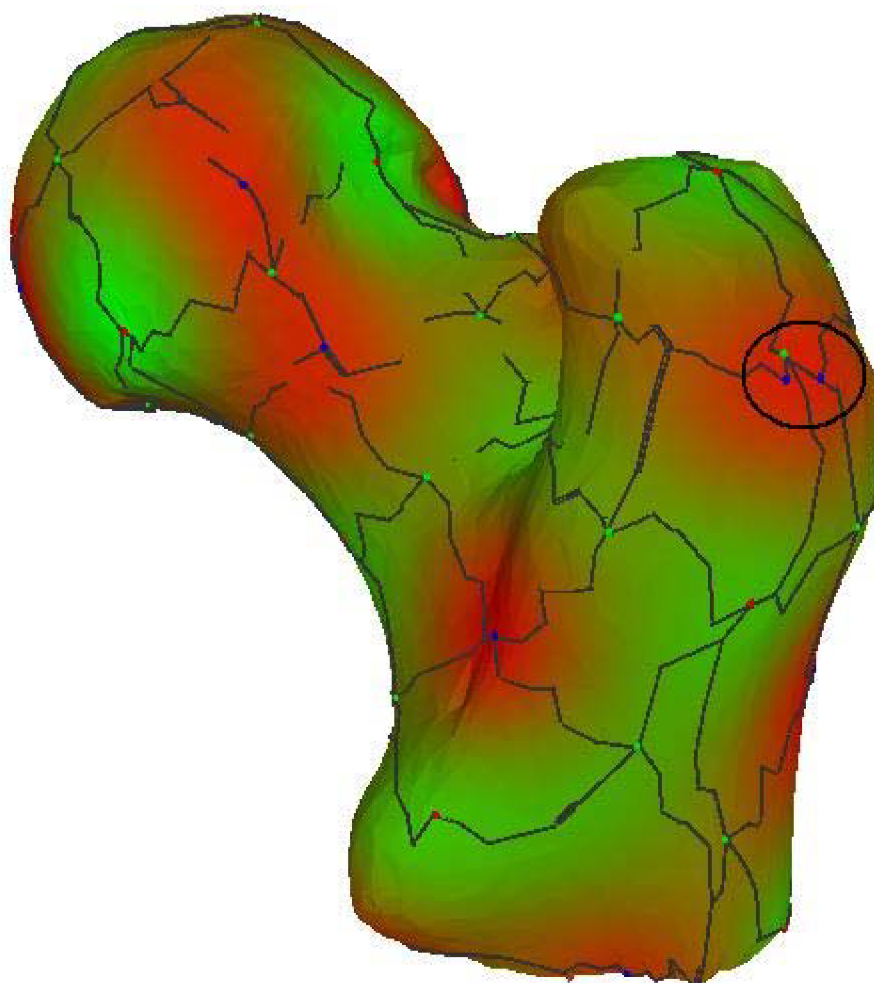
Obrázek 18: Model kloubu a jeho Morsův-Smalův komplex bez optimalizací. Nedostatky zvýrazněny.



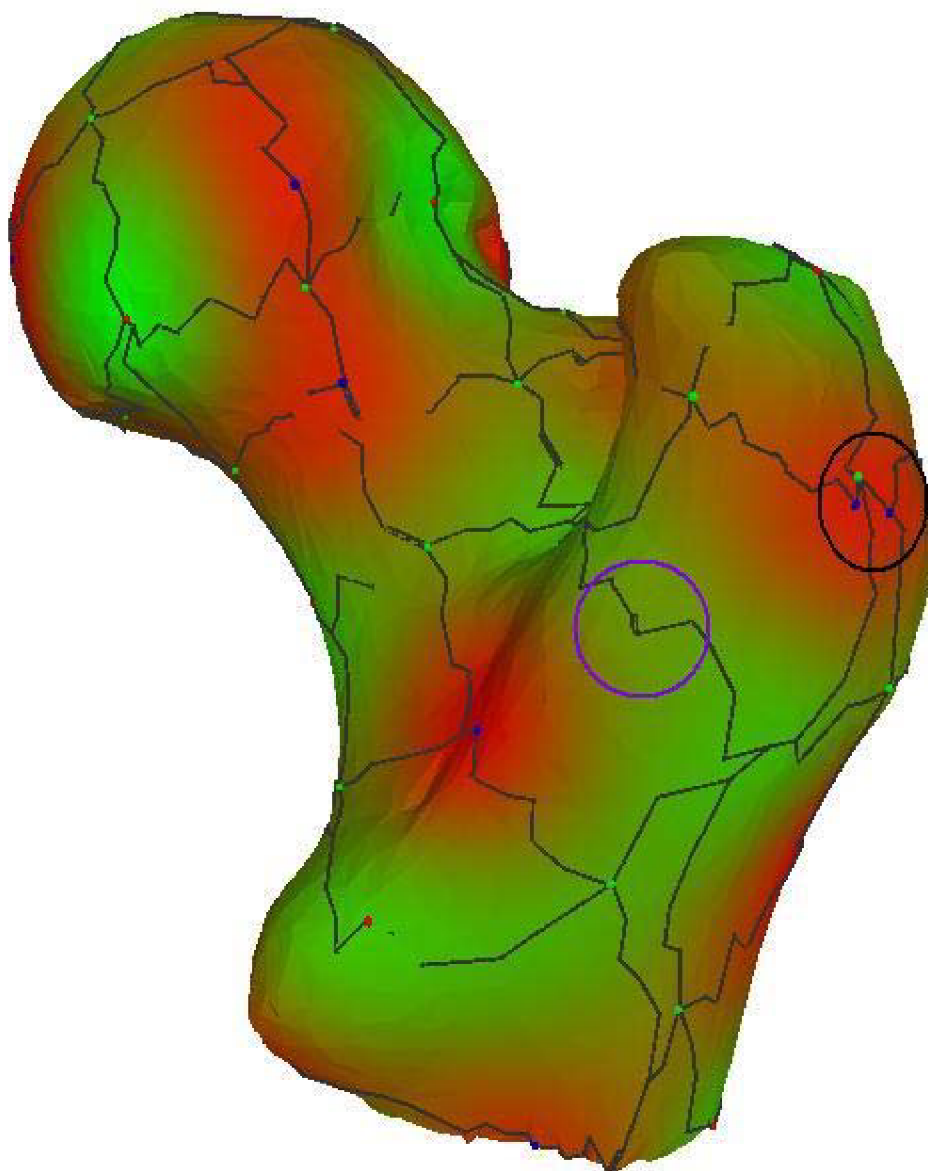
Obrázek 19: Nedostatky Quasi-Dual komplexu vytvořeného z neoptimalizovaného Morsova-Smalova komplexu.



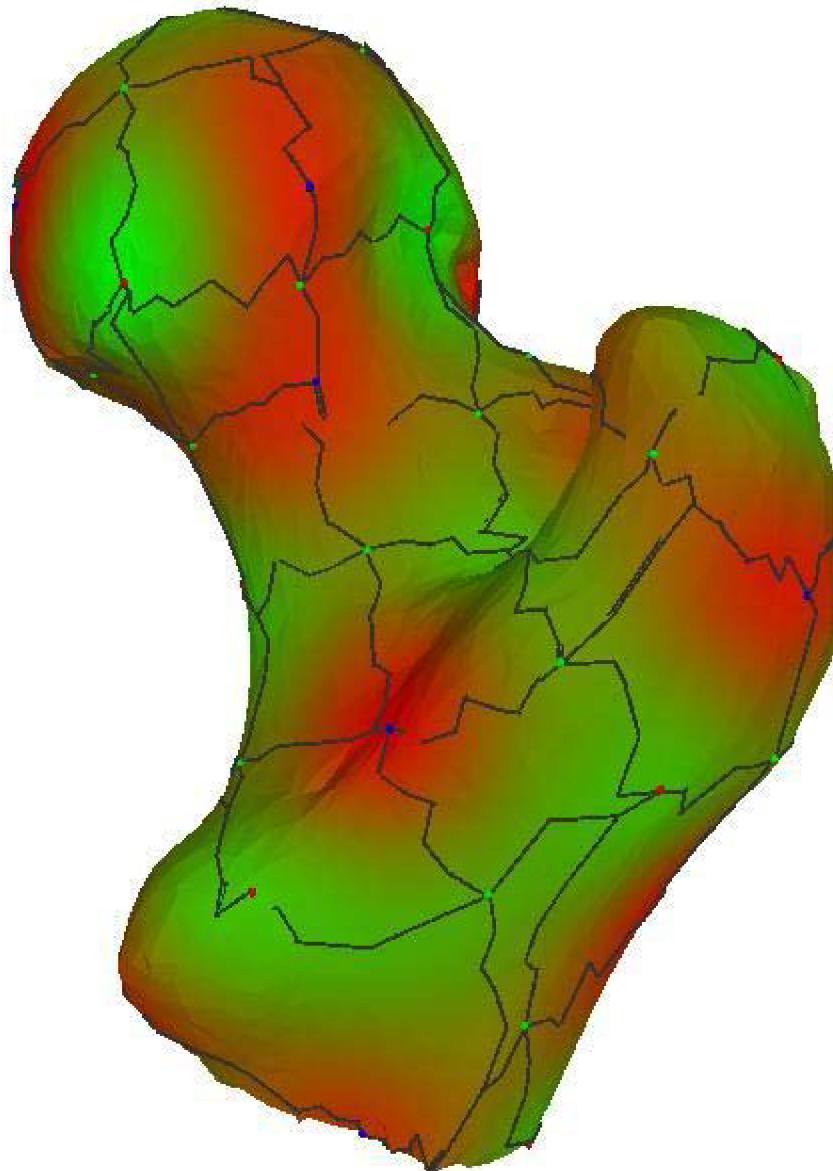
Obrázek 20: Model kloubu a jeho Morsův-Smalův komplex s odstraněním dvoumocných maxim a minim.
Zbývající nedostatky zvýrazněny.



Obrázek 21: Model kloubu a jeho Morsův-Smalův komplex s odstraněním dvoumocných maxim a minim. Zbývající nedostatky zvýrazněny.



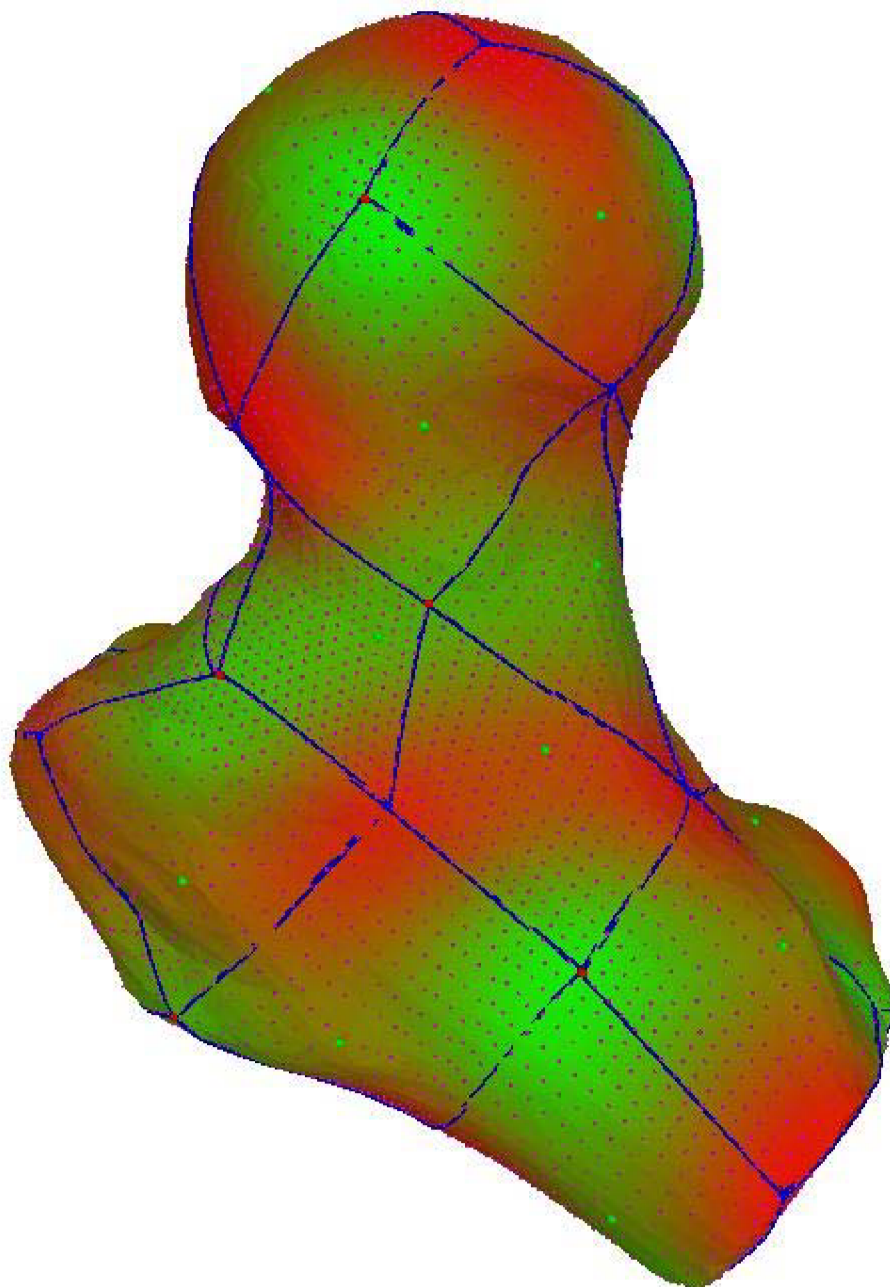
Obrázek 22: Model kloubu a jeho Morsův-Smalův komplex s odstraněním dvoumocných maxim a minim a použitím rušení na základě trvalosti s hodnotou 0.2 procent. Ačkoli došlo ke zrušení potřebného sedla, stále se vyskytují nedostatky.



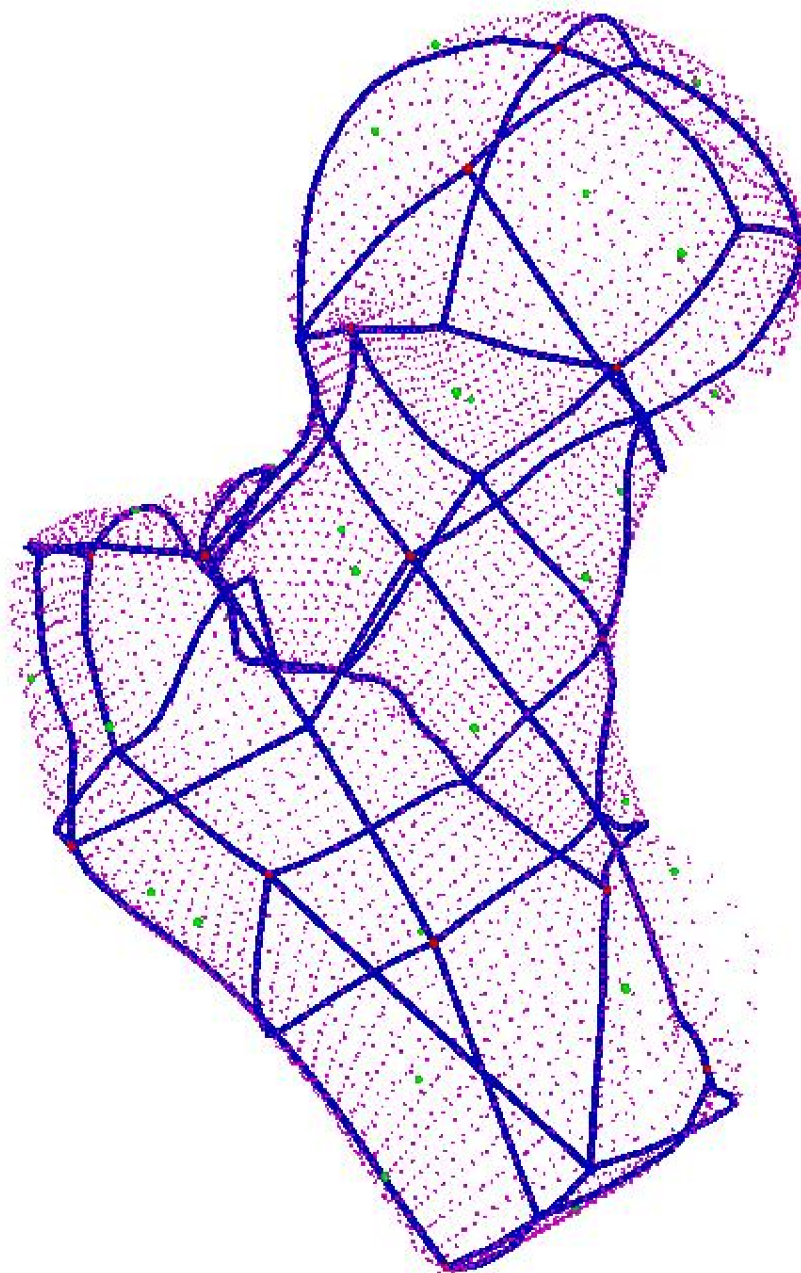
Obrázek 23: Model kloubu a jeho Morsův-Smalův komplex s odstraněním dvoumocných maxim a minim a použitím rušení na základě vzdálenosti s hodnotou 10 procent.

5.3 Quasi-Dual komplex a interpolační body

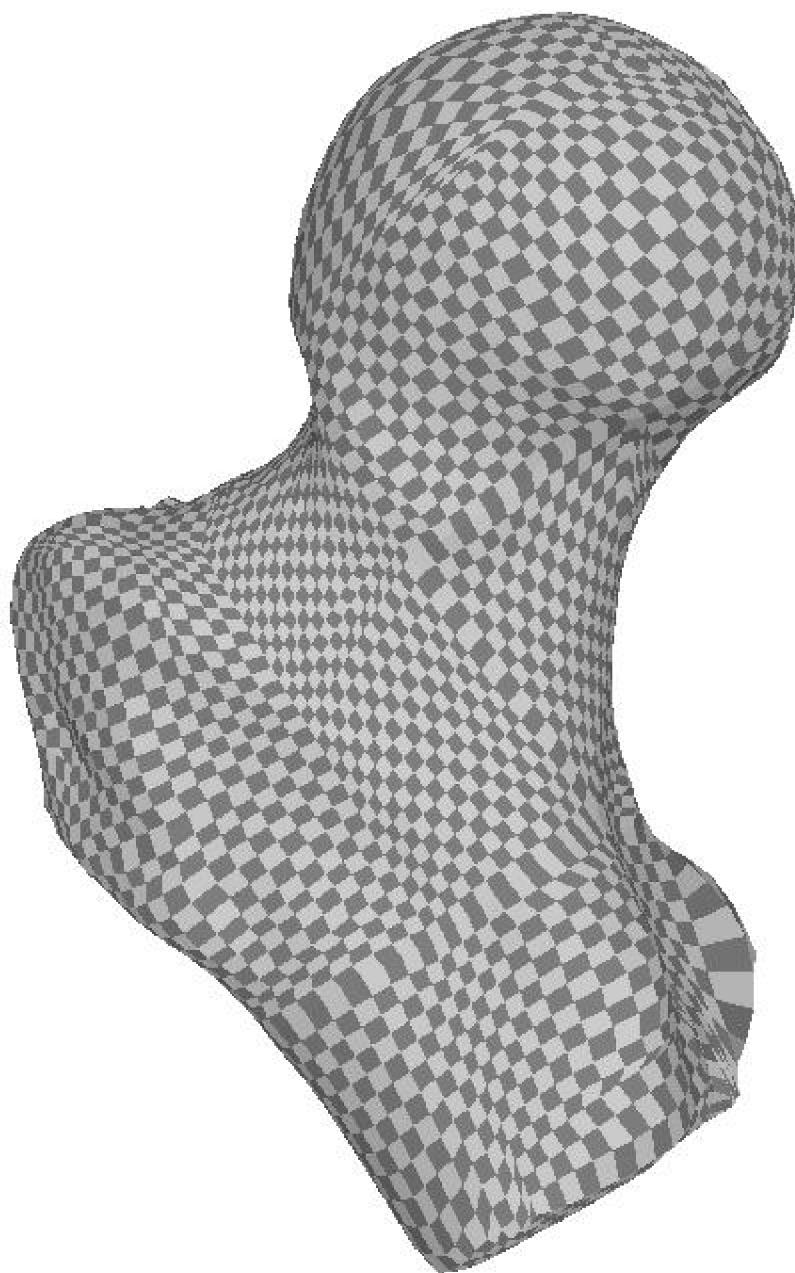
V této části si ukážeme výsledky výpočtů interpolačních bodů. Na obrázcích 24 a 25 můžeme vidět výsledný quasi-dual komplex a jeho síť interpolačních bodů pro 50. vlastní vektor. Dále také můžeme srovnat model kloubu vykreslený pomocí těchto interpolačních bodů s původním modelem z trojúhelníků (obr. 26 a 27 pro 50. vlastní vektor a obr. 28 a 29 pro 120. vlastní vektor).



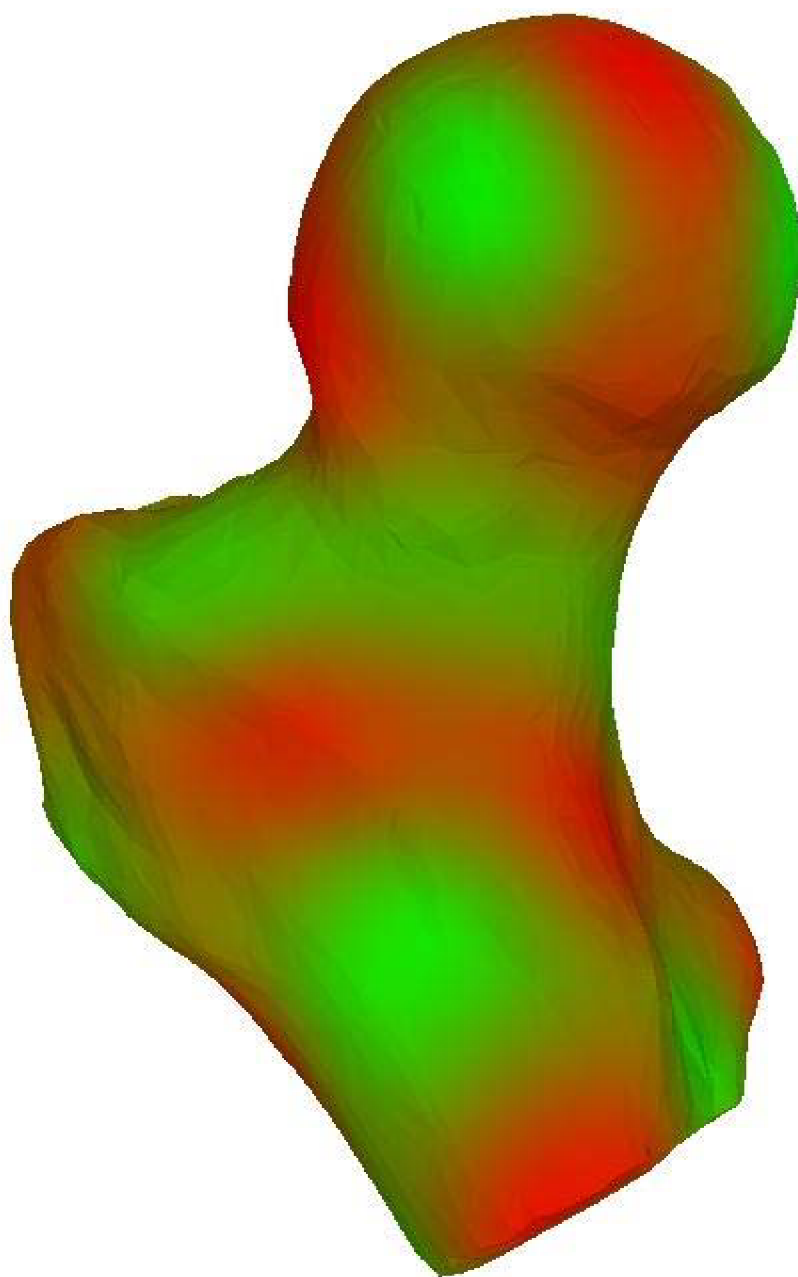
Obrázek 24: Model kloubu a jeho interpolační body.



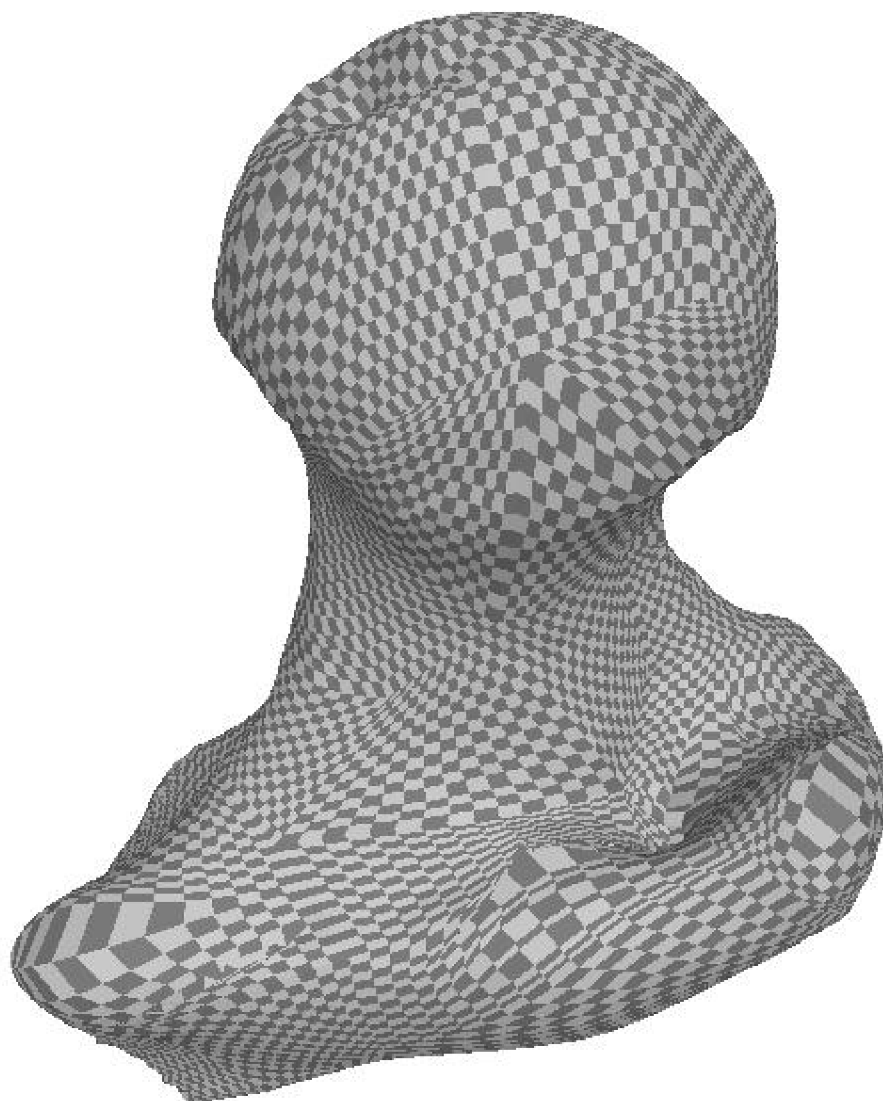
Obrázek 25: Síť interpolačních bodů.



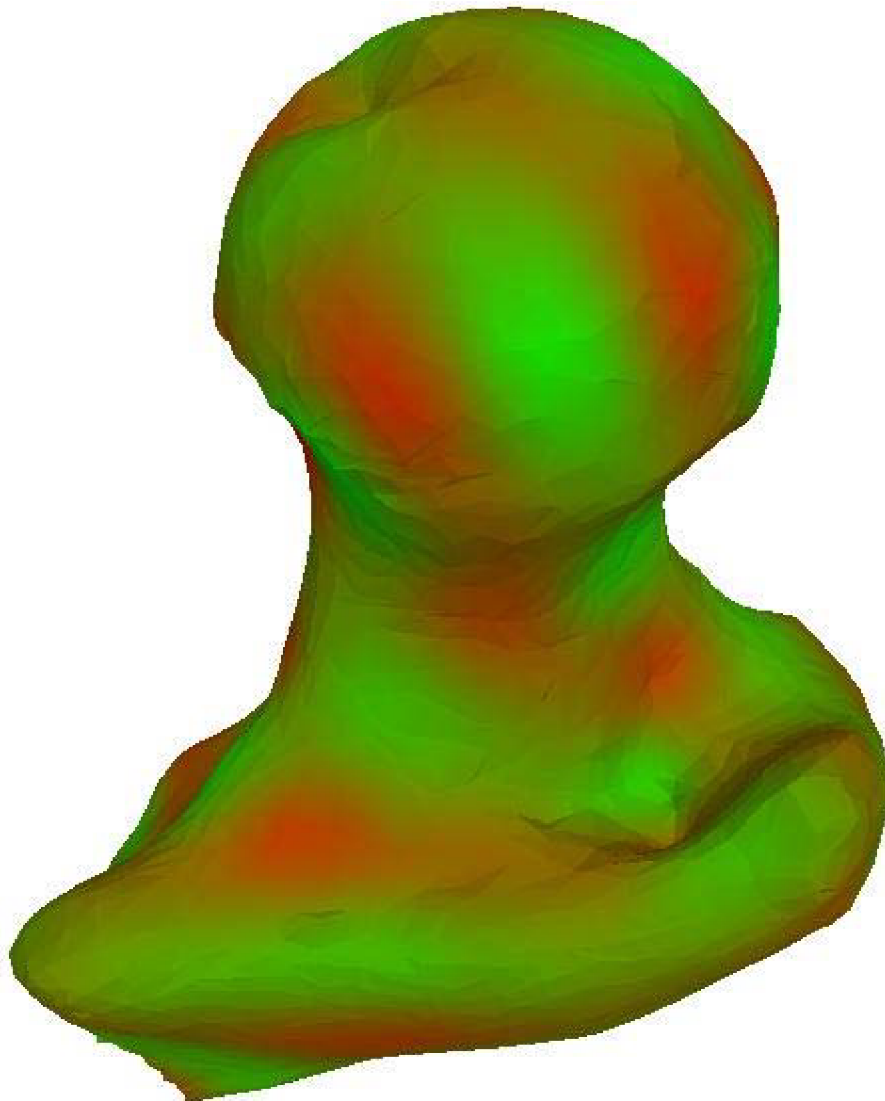
Obrázek 26: Model kloubu vykreslený z interpolačních bodů, použit 50. vlastní vektor.



Obrázek 27: Model kloubu vykreslený z trojúhelníků.

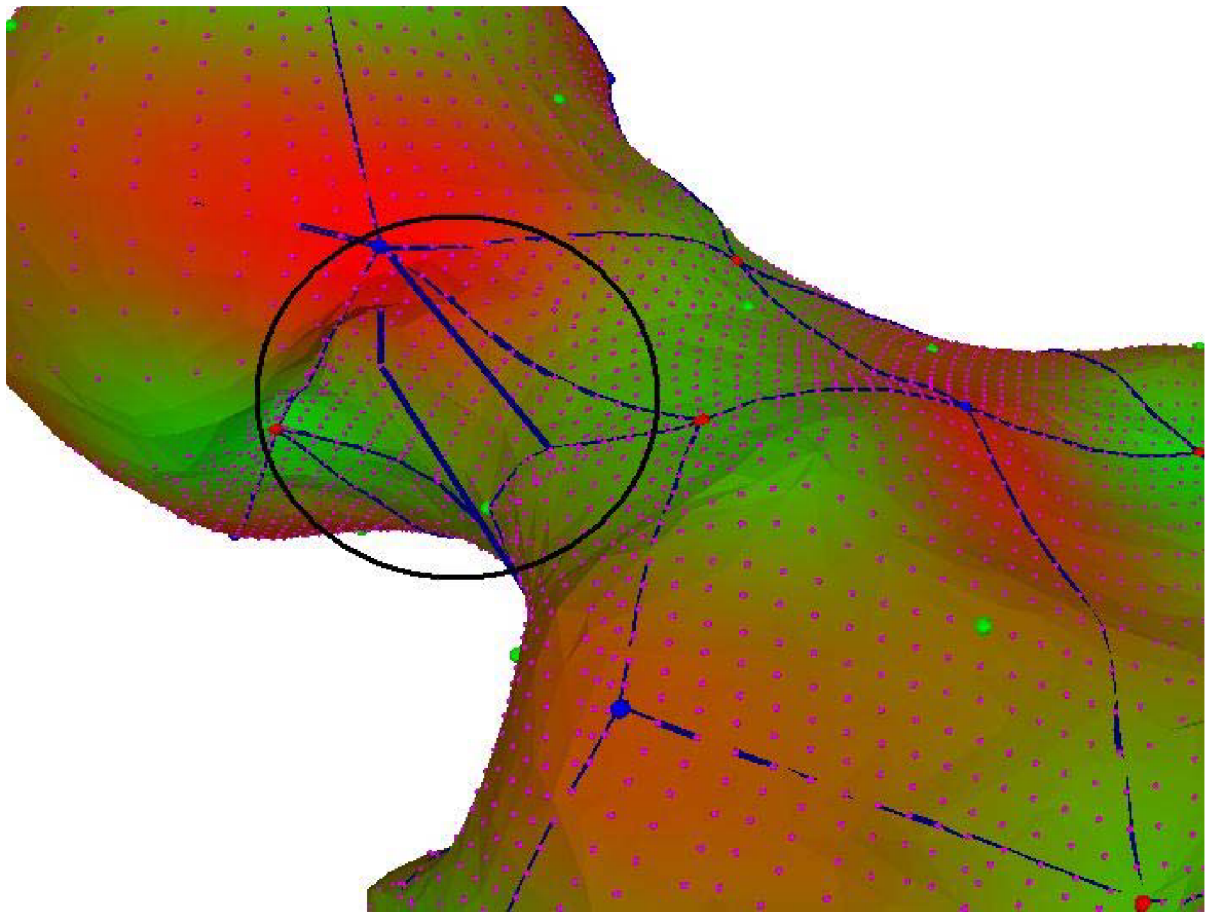


Obrázek 28: Model kloubu vykreslený z interpolačních bodů, použit 120. vlastní vektor.

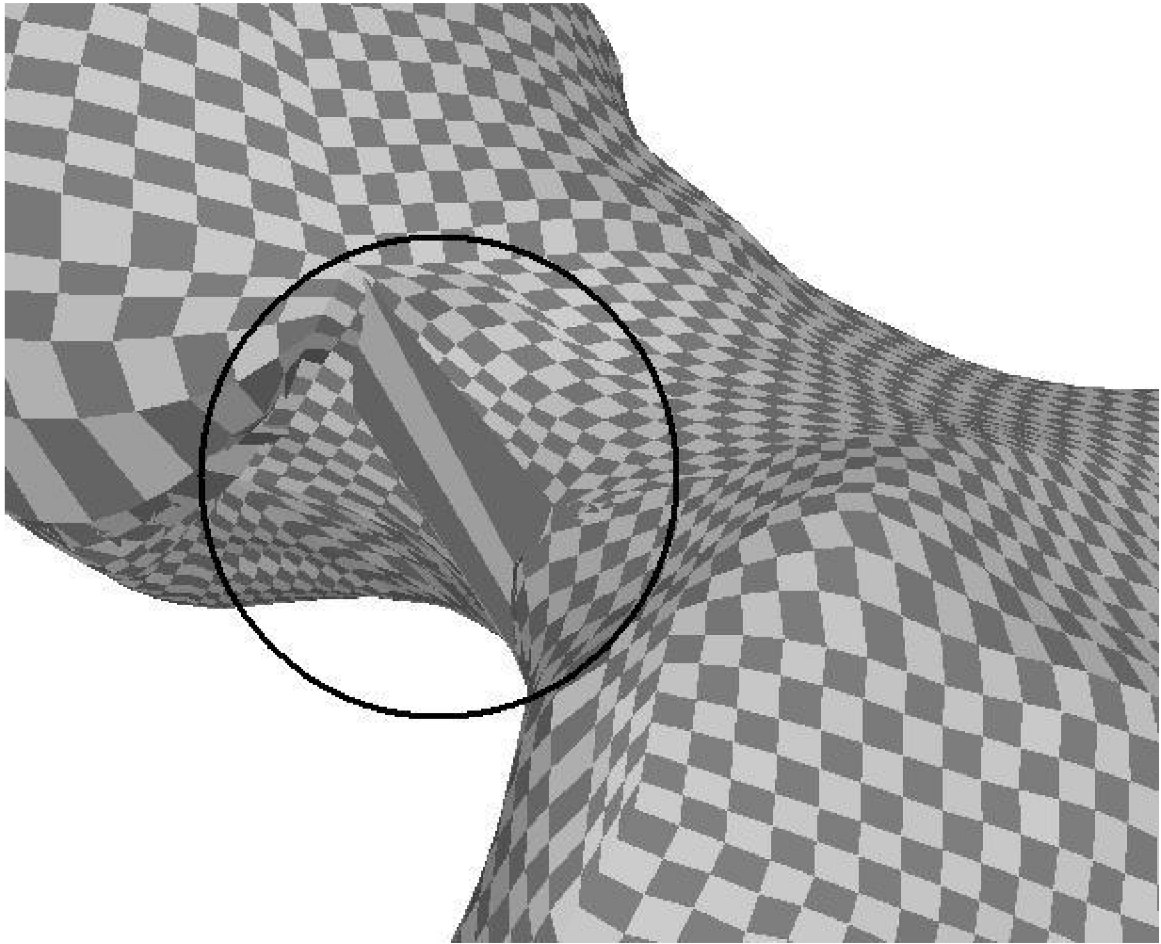


Obrázek 29: Model kloubu vykreslený z trojúhelníků.

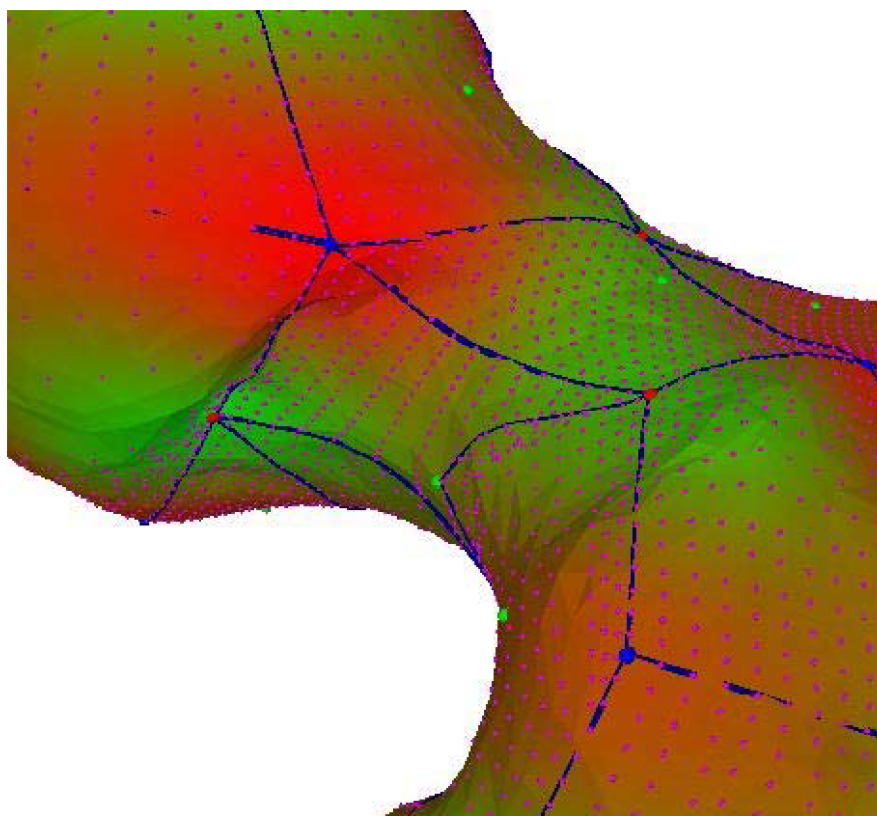
Nyní si ukážeme jak může zjemnění ovlivnit správnost výpočtu interpolačních bodů. Na obrázcích 30 a 31 zjemnění použito nebylo. Můžeme vidět, že v zakroužkovaných oblastech průsečíky dopočítány v nesprávné části modelu. Na obr. 32 a 33 vidíme nápravu použitím Zjemnění..



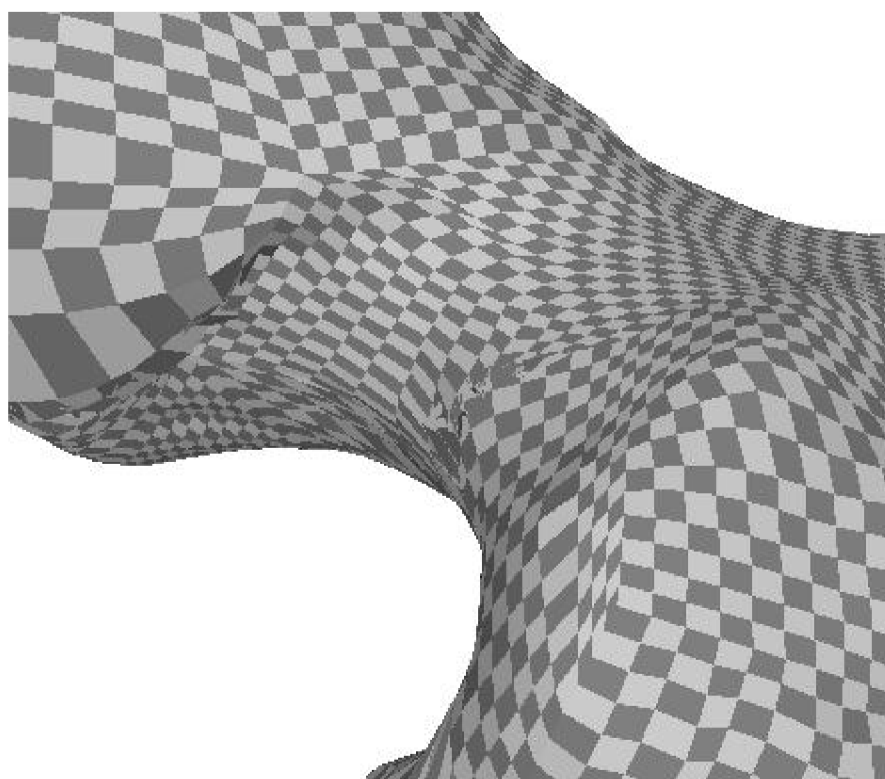
Obrázek 30: Chyba ve výpočtu interpolačních bodů. Nepoužito Zjemnění..



Obrázek 31: Chyba ve výpočtu interpolačních bodů. Nepoužito Zjemnění..

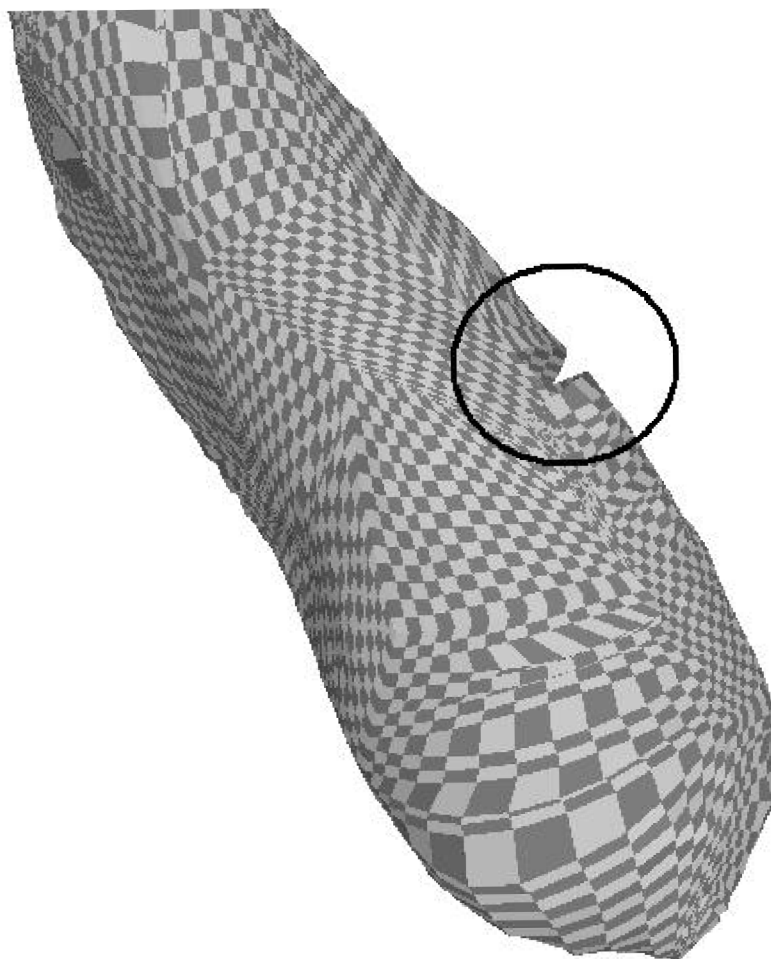


Obrázek 32: Odstranění chyby použitím Zjemnění..

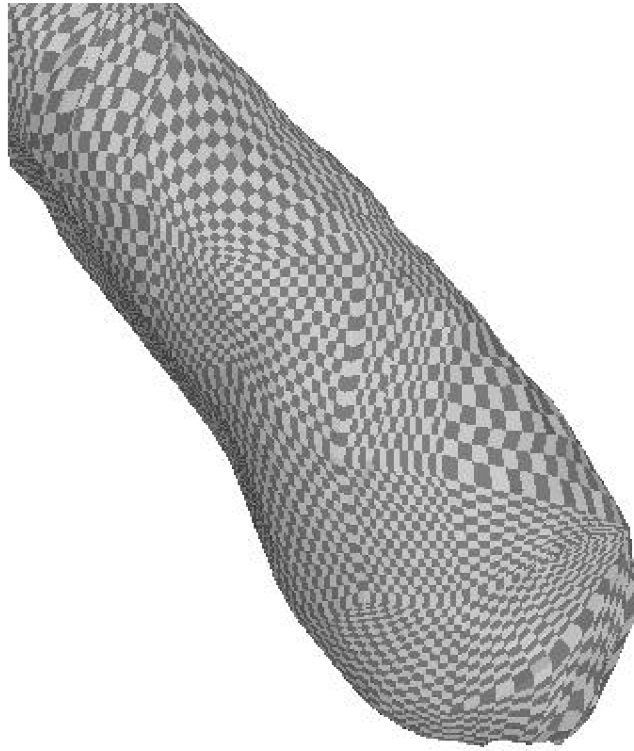


Obrázek 33: Odstranění chyby použitím Zjemnění..

Při zpracovávání modelu touto metodou, může nastat i případ, že výsledný quasi-dual komplex nepokrývá celý povrch modelu (obr. 34). Tato chyba se dá odstranit použitím vhodnějšího vlastního vektoru (obr. 35).



Obrázek 34: Chyba quasi-dual komplexu. Použit 50. vlastní vektor.



Obrázek 35: Odstranění chyby použitím 100. vlastního vektoru.

6 Závěr

Převod sítě na kvalitní čtyřúhelníkový základ je nejlepší cestou k převodu trojúhelníkových polygonálních 3D sítí na 3D spline plochy. Z tohoto čtyřúhelníkového základu lze již snadno dalšími metodami získat 3D spline plochy. V této práci jsem se zaměřil na co nejefektivnější získání semi-regulární čtyřúhelníkové sítě na základě spektrální analýzy. Dosažené výsledky touto metodou jsou uspokojivé a je možné je použít pro další práci s modelem. Ze získaných interpolačních bodů je možné dále dopočítat řídicí body pro aproximační spliny. Tato metoda není vhodná pro zpracování tvarově příliš složitých modelů, u kterých je vzniklý quasi-dual komplex nekvalitní, a je téměř nemožné dopočítat interpolační body.

Nevýhodou této metody je velmi časově náročný výpočet vlastních čísel a vlastních vektorů velmi rozsáhlých symetrických matic. Tento nedostatek jsem se pokoušel řešit použitím knihovny lapack, ale neúspěšně, pravděpodobně docházelo k příliš vysoké zaokrouhlovací chybě. Dalším možným řešením by bylo použití knihovny ARPACK, která by měla být dostatečně rychlá, přesná a stejně jako knihovna lapack umožňuje výpočet pouze několika potřebných vlastních vektorů.

Literatura

- [1] Kršek P. Dokumentace knihovny VectEntity2
- [2] Dong et al.: Quadrangulating a Mesh using Laplacian Eigenvectors, Technical Report No. UIUCDCS-R-2005-2583, 2005
- [3] Dong et al.: Spectral Surface Quadrangulation, 2006
- [4] Numerical Recipes in C, kapitola 11, <http://www.library.cornell.edu/nr/bookcpdf.html>
- [5] <http://www.fs.cvut.cz/cz/U201/map/math/teorie/mapt106.htm>

Příloha

Ovládání programů

Program **vypocet**:

Parametry

- v výpočet vlastních čísel a vlastních vektorů a jejich výpis do souboru, za tímto parametrem následuje jako další parametr název souboru s modelem ve formátu VectEntity2.
- n načte vlastní čísla a vektory ze souboru, který následuje jako druhý parametr a vypíše k vlastní vektor, kde k je třetí parametr.

Program **Prohlizec**:

Parametry

- n načte model ze souboru, který následuje jako druhý parametr a zpracuje ho podle vlastního vektoru, který je zadán jako třetí parametr, čtvrtý parametr udává vzdálenost (0 až 100 procent) pro topologickou optimalizaci.

Program se ovládá pomocí těchto kláves:

- 'q', 'x' a 'Esc' - ukončí program
- '1' - vykreslí se model se spektrem
- '2' - vykreslí se model se spektrem a s kritickými body
- '3' - vykreslí se model se spektrem a Morsův-Smalův komplex
- '4' - vykreslí se Morsův-Smalův komplex
- '5' - vykreslí se model se spektrem, quasi-dual komplex a interpolační body
- '6' - vykreslí se quasi-dual komplex a interpolační body
- '7' - vykreslí se model z interpolačních bodů
- '8' - vykreslí se model se spektrem a quasi-dual komplex
- '9' - vykreslí se quasi-dual komplex