

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Mantas Repečka

**Žmogaus sąkandžio fizikinio modelio sukūrimas pagal
tomografu išmatuotas dantų paviršių koordinates ir jo
tyrimas**

Magistro darbas

Darbo vadovas

prof. habil. dr. Rimantas Barauskas

Kaunas, 2012

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Mantas Repečka

**Žmogaus sąkandžio fizikinio modelio sukūrimas pagal
tomografu išmatuotas dantų paviršių koordinates ir jo
tyrimas**

Magistro darbas

Vadovas

prof. habil. dr. Rimantas Barauskas
2012-05-28

Recenzentas

doc. dr. Vytenis Punys
2012-05-28

Atliko

IFM-0/1 gr. stud.
Mantas Repečka
2012-05-28

Kaunas, 2012

Turiny

1.	ĮVADAS	6
2.	ANALITINĖ DALIS	7
2.1.	Modelio struktūra	7
2.2.	Modelio paviršiaus geometrija	8
2.3.	Paviršiaus glotninimo algoritmai	11
2.3.1.	Žymėjimai	11
2.3.2.	Metodologija	11
2.3.3.	Bezje paviršiaus lopinėlis. Teorija.	12
2.3.4.	Lokaliai kvadratinis paviršiaus interpoliavimas.....	13
2.3.5.	Lokalus paviršiaus interpoliavimas naudojant trečiojo laipsnio ribines kreives 15	
2.3.5.1.	Paviršiaus lopinėlių ribinių kreivių suradimas	15
2.3.5.2.	Ribų tangentinės plokštumos.....	16
2.3.5.3.	Paviršiaus lopinėlio sudarymas	17
2.4.	Baigtinių elementų metodas.....	19
2.4.1.	Netiesinis dinamikos uždavinys. Bendroji formuluotė.	19
2.4.2.	Apkrovos priklausomybė nuo būvio kintamųjų.	21
3.	PROJEKTINĖ DALIS	23
3.1.	Projektavimo algoritmo struktūra	23
3.2.	Geometrijos netobulumų šalinimas.....	23
3.3.	Kremzlinių diskų sukūrimas	25
3.4.	Paviršiaus glotninimas	26
3.5.	Paviršiaus padengimas trianguliaciniu tinkleliu ir kremzlinių diskų tūrio užpildymas tetraedrais	28
3.6.	Keyword failų sudarymas	28
4.	Rezultatai	30
5.	IŠVADOS	32
6.	LITERATŪROS SąRAŠAS	33

Žmogaus sąkandžio fizikinio modelio sukūrimas pagal tomografu išmatuotas dantų paviršių koordinates ir jo tyrimas

Santrauka

Šio tyrimo tikslas buvo ištirti slėgio jėgų pasiskirstymą žmogaus smilkininio apatinio žandikaulio sąnariuose diskuose, kurios atsiranda sukandimo metu.

Darbe pateikiamas biomechaninis baigtinių elementų modelis, kuriame sąveikauja apatinio ir viršutinio žandikaulio dantų arkos ir smilkininio apatinio žandikaulio sąnariniai diskai. Modelio apdorjami duomenys buvo gauti iš kompiuterinio tomografo. Tiriamasis modelis susideda iš tokiu dalių: dviejų kietų (kaulinių) struktūrų kuriom priklauso apatinio ir viršutinio žandikaulio dantų arkos, du apatinio žandikaulio sąnariai, dvi smilkinkaulio žandikaulinės duobutės ir du kremzlinio audinio sąnariniai diskai. Modelyje viršutinio žandikaulio dantų arka buvo įtvirtinta erdvėje. Apatinio žandikaulio dantų arka gali judėti erdvėje sinchroniškai su apatinio žandikaulio sąnariais, veikiant apatinio žandikaulio raumenų jėgai. Toks judesys sukelia kontaktą tarp dantų paviršių ir kremzliniuose diskuose, kurie randasi tarp apatinio žandikaulio sąnarių ir žandikaulinių duobučių. Modelio skaičiavimai buvo atlikti su LS-DYNA baigtinių elementų programine įranga.

Gauti rezultatai pateikiami 3D vaizdu, kuriame atsispindi slėgio jėgos kremzliniuose diskuose. Modelio praktinė nauda yra ta, jog padeda įvertinti atsirandančias slėgio jėgas kremzliniuose diskuose, priklausomai nuo to koks pas žmogų yra sukandimas.

Human occlusion physical model creation and study according to teeth surface coordinates measured by computer tomograph.

Summary

The objective of this study was to investigate distribution of stresses in the human TMJ discs, generated during clenching.

The work presents a biomechanical finite element model of interaction of mandibular and maxillary dental arches and the TMJ discs of a particular person, based on real geometrical data obtained from spiral computed tomography two-dimensional images. 3D contour coordinates – point clouds were collected from these images and solid model was created. The system under investigation consisted of eight basic parts: two rigid structures representing the mandibular and maxillary dental arches, two mandibular condyles, two mandibular fossae of temporal bone, and solid models of two articular discs. The model of maxillary dental arch was fixed in space. The model of the mandibular dental arch was able to move in space synchronically with the mandibular condyles under action of applied forces, which were considered as prescribed and known at insertion points of masticatory muscles. The motion of the mandible was constrained by interdental contact interactions and contact interaction with articular discs, which were situated in between mandibular condyles and mandibular fossae of temporal bone. The model was implemented by using LS-DYNA finite element software.

The obtained results presented a 3D view of stresses exhibited in the articular discs with values of interaction forces. The expected practical value of the developed model is the facilitation of biomechanical evaluations of the influence of tolerances of teeth shapes and occlusal areas together with the supporting areas on the final stress distribution in articular discs.

1. ĮVADAS

Dantų biomechaninės funkcijos paprastai sukelia slėgio ir įtempimo jėgas apatinio žandikaulio sąnariuose diskuose. Geresnis supratimas apie atsirandančias tokias jėgas padeda diagnozuoti ir gydyti stomatologines ligas. Deja be intervencinio būdo, slėgis nepamatuojamas. Tyrinėjimai šia tema yra riboti, todėl kad anatomicinės dalis sunku pasiekti ir prietaisais kurie galėtų pamatuoti įtempimus gali sukelti audinių pažeidimus.

Pastarąjį dešimtmetį baigtinių elementų metodas tapo galingu analizės instrumentu tyrinėjant biomechanikos procesus. Baigtinių elementų programose pasitelkiant išreikštinių dinaminį būdą, galima sėkmingai modeliuoti tokias sistemas. Šitame darbe buvo pasirinkta LS-DYNA baigtinių elementų programa. Išreikštinės dinamikos programa simuliuoja procesą laike, ekstrapolijuojant sistemos būseną kitame laiko žingsnyje, pagal esama sistemos būseną, su pritaikomomis atitinkamomis jėgomis. Skaičiavimai yra atliekami žingsnis po žingsnio. Paprastai, tokie modeliavimai pateikia įdomios informacijos apie modeliuojamos sistemos elgseną ir būseną, kurių negalima išmatuoti tiesiogiai. Kita vertus norint pasitikėti gautais modeliavimo rezultatais, simuliacijoje naudojamus duomenys reikia apdoroti taip, kad jie taptų kiek galima labiau panašūs į realią sistemą.

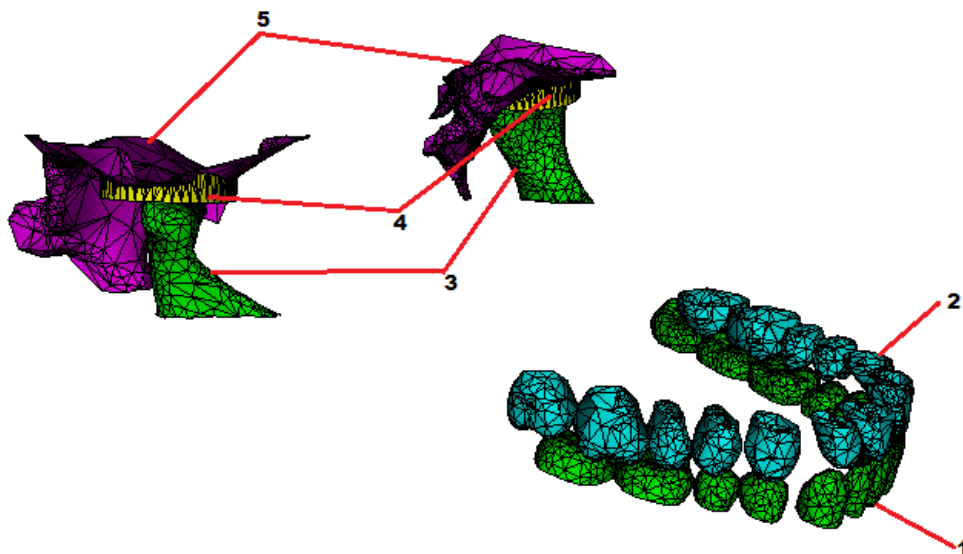
Šio darbo pagrindiniai tikslai yra:

- Sukurti biomechaninį žmogaus apatinio/viršutinio žandikaulio ir žandikaulio sąnarių kremzlių 3D modelį.
- Išmatuoti slėgio jėgas atsirandančias apatinio žandikaulio sąnariuose diskuose, sukandimo metu.

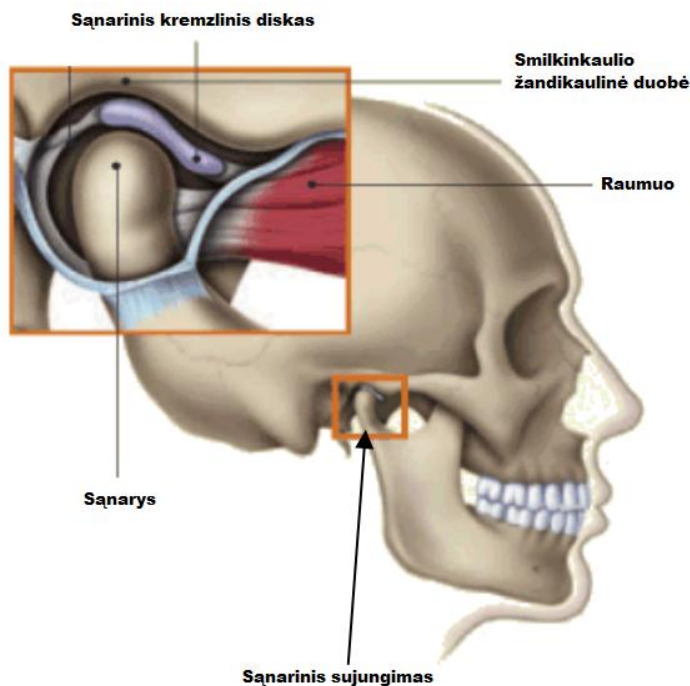
2. ANALITINĖ DALIS

2.1. Modelio struktūra

Tiriamąjį modelį duomenys gauti iš kompiuterinio tomografo ir paversti į 3D modelį. Modelis yra suskirstytas į kelias anatomines dalis ir aptrauktas trikampiniu tinkleliu. Modelio sandara pavaizduota 1 paveiksle.



1 pav. Modelio sandara: 1) apatinio žandikaulio dantų arka, 2) viršutinio žandikaulio dantų arka, 3) apatinio žandikaulio sąnariai, 4) smilkininio apatinio žandikaulio sąnariniai diskai, 5) smilkinkaulio žandikaulinės duobės



2 pav. Modelio anatominė struktūra.

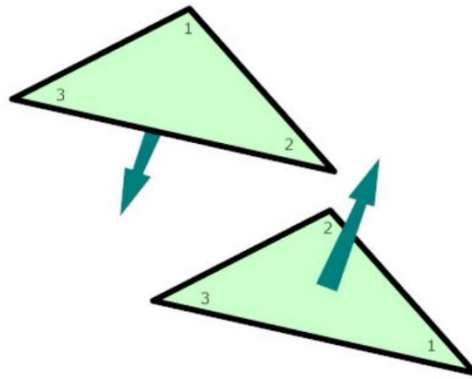
Šiame darbe svarbiausia anatomicinė dalis yra apatinio žandikaulio sąnariniai kremzliniai diskai. Šie diskai randasi apatinio žandikaulio sąnaryje ir yra atsakingi už apatinio žandikaulio judėjimą. Vykstant sukandimui burnos raumenys per kramtymo sistemos dalis perduoda jėgas į diskus ir sukelia tempimo jausmą šioje dalyje. Jausmas gali sustiprėti ir kelti nemalonius jausmus dėl pasikeitusios danties formos (suprastėja sukandimas).

2.2. Modelio paviršiaus geometrija

Kad modelį būtų galima atvaizduoti 3D paviršiumi, kompiuterinio tomografo duomenys buvo išsaugoti STL formatu. STL failo formatas tai trianguliacinis 3D paviršiaus geometrijos atvaizdavimo būdas. Paviršius mozaikiškai padengiamas orientuotų trikampių aibe. Kiekvienas trikampis yra apibūdinamas išorės normale ir trimis taškais sudėliotais prieš laikrodžio rodyklę, kurie atspindi trikampio viršūnes. Trikampių briaunų santykis ir orientacija priklauso nuo paviršiaus išlenkimų. Trikampių dydis apsprendžia perteikiamos geometrijos paviršiaus kokybę. Ar objektas padengtas tinkamo dydžio trikampaiais, priklauso nuo objekto formos. Plokštumos gali būti padengtos stambiais trikampaiais ir toks padengimas gali būti laikomas kokybišku, nes tiksliai atspindi nuskenuota realų objektą. Objektai kurie turi išlenkimų, reikalauja smulkesnio (tankesnio) trikampių paviršiaus tinklo, kadangi tankesnis trikampių kiekis tiksliau aproksimuoja paviršiaus išlenkimus. Tiriamojo modelio STL duomenys yra vidutinės kokybės, todėl tolesniuose skyriuose aptarsime kaip pagerinti esamo modelio trianguliacinį paviršių.

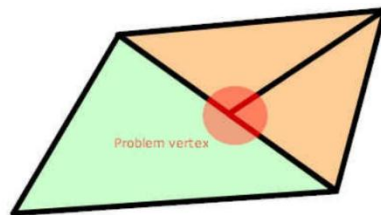
STL formatas turi atitikti tokius reikalavimus:

- Normalė ir kiekviena trikampio viršūnė yra apibūdinama trimis koordinatėmis. Taigi yra saugoma 12 skaičių kiekvienam trikampiui.
- Kiekvienas trikampis yra ribos dalis tarp objekto vidaus ir išorės. Trikampio orientacija (kuris yra „viduje“ ir kuris yra „išorėje“) yra apsprendžiama dviem veiksmiais. Pirma, normalės kryptis turi būti nukreipta į išorę. Antra, trikampių viršūnės turi būti aprašomos prieš laikrodžio rodyklę, jei žiūrime į objekto išorę (dešinės rankos taisyklė).



3 pav. Trikampio viršūnių išdėstymas ir normalės kryptys.

- Kiekvienas trikampis turi dalintis dviem viršūnėmis su kiekvienu greta esančiu trikampiu (viršūnė su viršūne taisyklė). Žemiau pateiktas pavyzdys, su netinkamu viršūnių pasidalijimu.



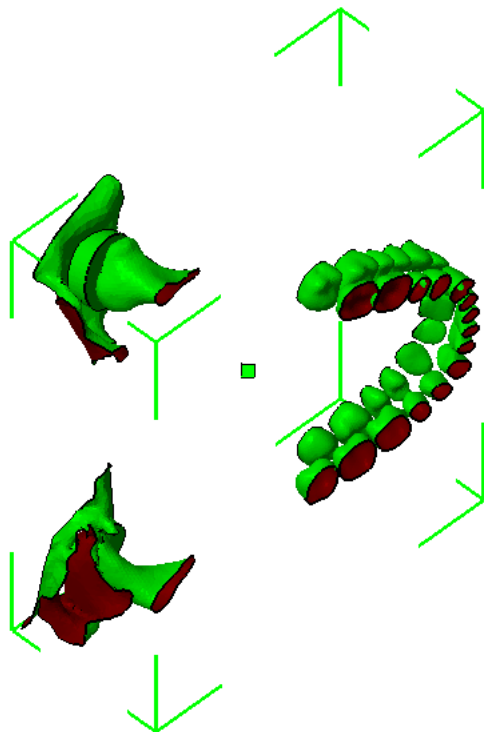
4 pav. Netinkamas trikampiu viršūnių pasidalijimas.

STL standartas gali būti dviejų formatų – ASCII ir dvejetainis. ASCII formato struktūra lengvai nuskaitoma paprasto tekstinio redaktoriaus. Dvejetainis formatas dažniausiai labiau naudojamas, nei ASCII, dėl to, kad ASCII formato failai užima gerokai daugiau vietos. ASCII formato geometrijos aprašas pradedamas žodžiu „solid“. Paskutinėje aprašo eilutėje užrašoma „endsolid“. Tarp pastarųjų eilučių aprašoma visa objekto geometrija (trikampių aibės). Sintaksės pavyzdys:

```
solid
:
:
facet normal 0.0 0.0 1.0
outer loop
vertex 1.0 1.0 0.0
vertex -1.0 1.0 0.0
vertex 0.0 -1.0 0.0
endloop
endfacet
:
:
endsolid
```

STL failai nesaugo informacijos apie objektų matavimo vienetus. Programa kuri nuskaito STL failus pati apsprendžia matavimo vienetus. Pavyzdžiui, jei x/y/z reikšmės dydis mažiau nei 10, visko gali būti jog programa priskirs matavimo vienetus coliais.

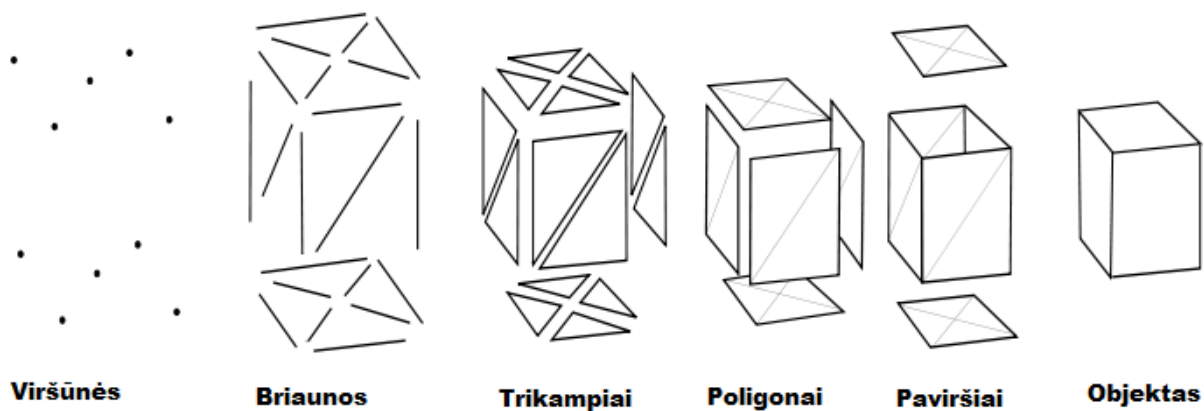
Visa STL geometrija yra patalpinama į ribinę dėžę (angl. boundary box).



5 pav. Ribinė geometrijos dėžė.

Ribinė dėžė parodo STL geometrijos ribas ir koordinacių atskaitos pradžios tašką. X/Y/Z atskaitos taškas gali būti nebūtinai (0;0;0). Visa dėžę galime pastumti kuria norime kryptimi.

Apskritai darbe naudojamos geometrijos paviršius susideda iš tokių dalių:



6 pav. Geometrinio objekto skaidymas į smulkesnes sudėtinės dalis.

2.3. Paviršiaus glotninimo algoritmai

Modelio paviršiaus aptraukimas tinklelių (trianguliacinis procesas) yra labai svarbus baigtinių elementų metodo skaičiavimuose. Esamas paviršius kol kas turi vidutinės kokybės paviršiaus padengimą. Kokybiškai padengti paviršių yra svarbu todėl, kad modelis tampa panašesnis į realų modelį. Dabartinis paviršius atrodo turintis per aštirus, kampuotus išlenkimus. Tai nėra pageidaujama savybė, kadangi realiai dantų paviršius dažniausiai būna glotnus. Taip pat, kampuotumai paviršiuje įneša netikslumų skaičiavimuose, kadangi atsiranda prastesnis sukandimas. Problema bandysime spręsti pasitelkę paviršiaus glotninimo algoritmus.

Kad sukurti aukšto detalumo paviršių iš esamo šiurkštaus paviršiaus, reikia sukurti ryšį tarp trikampio viršūnių detaliame paviršiuje ir šiurkščiame paviršiuje. Idėja yra pažiūrėti kur trikampio viršūnės detaliame paviršiuje atsidurs lyginant atitinkamai su šiurkštaus trikampio viršūnių pozicijom. Kad tai įgyvendinti reikia interpoliuoti šiurkštaus paviršiaus trikampių tinklelio viršūnes.

2.3.1. Žymėjimai

Tegu $\{T_d\}_d$ būna viršūnių trianguliacija detaliame tinklelyje ir $\{T_c\}_c$ viršūnių trianguliacija šiurkščiame tinklelyje. v_d ir v_c pažymime kaip aibe viršūnių iš $\{T_d\}$ ir $\{T_c\}$ atitinkamai. Naujos viršūnės po interpoliacijos yra pažymimos v'_c ir v'_d atitinkamai. Viršūnės detaliame tinklelyje yra pažymimos didžiosiomis raidėmis, o šiurkščiame tinklelyje viršūnės žymimos mažosiomis raidėmis.

$$v_d = \{\mathbf{V}_1, \dots, \mathbf{V}_N\}$$

$$v_c = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$$

$$N \gg n$$

Jei kalbama apie vieną trikampį tinklelyje tuomet trikampio viršūnės žymime x_0, x_1 ir x_2 , ir atitinkamai šių viršūnių paviršiaus normalės vektorius pažymime N_0, N_1, N_2 . Mes naudosime $P_{i,j,k}$ kad pažymėtume kontrolinius taškus trianguliaciniame Bezje lopinėlyje $P_{i,j,k} \in \mathbb{R}^3$.

2.3.2. Metodologija

Mes norime susieti detaliojo tinklelio trikampių viršūnių pozicijas kartu su šiurkštaus tinklelio trikampio viršūnių pozicijom. Tuomet mums reikia paskaičiuoti pozicijas viršūnių v'_d nuo viršūnių v'_c . Paprasčiausias būdas tai padaryti, būtų apibūdinti kiekviena viršūnę detaliame

tinklelyje, ortogonaliai projektuojant taškus į artimiausias šiurkščiojo tinklelio trikampio viršūnes. Tegu V būna viršūnė detaliame tinklelyje ir v_1, v_2, v_3 būna artimiausios viršūnės V viršūnei šiurkščiam tinklelyje. Dabar V galima išreikšti kaip sumą taškų trikampio plokštumoje ir vektoriaus statmenį į šią plokštumą.

$$\mathbf{V} = \mathbf{v}_1 + s(\mathbf{v}_2 - \mathbf{v}_1) + t(\mathbf{v}_3 - \mathbf{v}_1) + u((\mathbf{v}_2 - \mathbf{v}_1) \times (\mathbf{v}_3 - \mathbf{v}_1)).$$

Tuomet galima paskaičiuoti V'

$$\mathbf{V}' = \mathbf{v}'_1 + s(\mathbf{v}'_2 - \mathbf{v}'_1) + t(\mathbf{v}'_3 - \mathbf{v}'_1) + u((\mathbf{v}'_2 - \mathbf{v}'_1) \times (\mathbf{v}'_3 - \mathbf{v}'_1)).$$

Mūsų pagrindinis tikslas yra interpoliuoti paviršiaus lopinėlių $f_T(v, w)$ kiekvienam trikampiui T trianguliacijoje $\{T_c\}$. Tegu T_c būna artimiausias trikampis, viršūnei V detaliame tinklelyje. Tokiu pačiu būdu kaip aukščiau, išreiškiame V kaip tašką paviršiaus lopinėlyje T_c , taipogi vektorių statmena paviršiui.

$$\mathbf{V} = f_{T_c}(s, t) + u \left(\frac{\partial f_{T_c}(s, t)}{\partial v} \times \frac{\partial f_{T_c}(s, t)}{\partial w} \right).$$

Interpoliuojant paviršiaus lopinėlių tokiu pačiu būdu kaip trikampi $T'_c, f_{T'_c}(v, w)$, mes galime apskaičiuoti V'

$$\mathbf{V}' = f_{T'_c}(s, t) + u \left(\frac{\partial f_{T'_c}(s, t)}{\partial v} \times \frac{\partial f_{T'_c}(s, t)}{\partial w} \right).$$

2.3.3. Bezje paviršiaus lopinėlis. Teorija.

Bezje trikampis laipsniu n , yra paviršius aprašytas taip

$$\mathbf{S}(u, v, w) = \sum_{i+j+k=n} \mathbf{P}_{i,j,k} \frac{n!}{i!j!k!} u^i v^j w^k = \sum_{i+j+k=n} \mathbf{P}_{i,j,k} b_{i,j,k}^n(u, v, w), \quad i, j, k \geq 0.$$

Kur (u, v, w) yra baricentrinės koordinatės.

Dalinė išvestinė S , u atžvilgiu

$$\begin{aligned} \frac{\partial \mathbf{S}(u, v, w)}{\partial u} &= \sum_{i+j+k=n, i \geq 1} \mathbf{P}_{i,j,k} \frac{n!}{i!j!k!} i u^{i-1} v^j w^k = n \sum_{i+j+k=n-1} \mathbf{P}_{i+1,j,k} \frac{(n-1)!}{i!j!k!} u^i v^j w^k \\ &= n \sum_{i+j+k=n-1} \mathbf{P}_{i+1,j,k} b_{i,j,k}^{n-1}(u, v, w). \end{aligned}$$

Analogiškai v ir w atžvilgiu būtu

$$\frac{\partial \mathbf{S}(u, v, w)}{\partial v} = n \sum_{i+j+k=n-1} \mathbf{P}_{i,j+1,k} b_{i,j,k}^{n-1}(u, v, w)$$

$$\frac{\partial \mathbf{S}(u, v, w)}{\partial w} = n \sum_{i+j+k=n-1} \mathbf{P}_{i,j,k+1} b_{i,j,k}^{n-1}(u, v, w).$$

Pasinaudojant šiomis formulėmis, surandame kryptinę išvestinę išilgai baricentrinės krypties $\mathbf{d} = (d, e, f)$

$$\frac{\partial \mathbf{S}(u, v, w)}{\partial \mathbf{d}} = n \sum_{i+j+k=n-1} (d\mathbf{P}_{i+1,j,k} + e\mathbf{P}_{i,j+1,k} + f\mathbf{P}_{i,j,k+1}) b_{i,j,k}^{n-1}(u, v, w).$$

Mums kryptinę išvestinę ribose $u = 0$, $v = 0$ ir $w = 0$ yra specialaus svarbumo, kryptyje statmenai ribai $(1, -1/2, -1/2)$, $(-1/2, 1, -1/2)$ $(-1/2, -1/2, 1)$.

$$\frac{\partial \mathbf{S}(0, 1-w, w)}{\partial (1, -1/2, -1/2)} = n \sum_{j+k=n-1} \left(\mathbf{P}_{1,j,k} - \frac{1}{2} (\mathbf{P}_{0,j+1,k} + \mathbf{P}_{0,j,k+1}) \right) b_{0,j,k}^{n-1}(0, 1-w, w)$$

$$\frac{\partial \mathbf{S}(u, 0, 1-u)}{\partial (-1/2, 1, -1/2)} = n \sum_{i+k=n-1} \left(\mathbf{P}_{i,1,k} - \frac{1}{2} (\mathbf{P}_{i+1,0,k} + \mathbf{P}_{i,0,k+1}) \right) b_{i,0,k}^{n-1}(u, 0, 1-u)$$

$$\frac{\partial \mathbf{S}(1-v, v, 0)}{\partial (-1/2, -1/2, 1)} = n \sum_{i+j=n-1} \left(\mathbf{P}_{i,j,1} - \frac{1}{2} (\mathbf{P}_{i+1,j,0} + \mathbf{P}_{i,j+1,0}) \right) b_{i,j,0}^{n-1}(1-v, v, 0).$$

Suprastintos formulės:

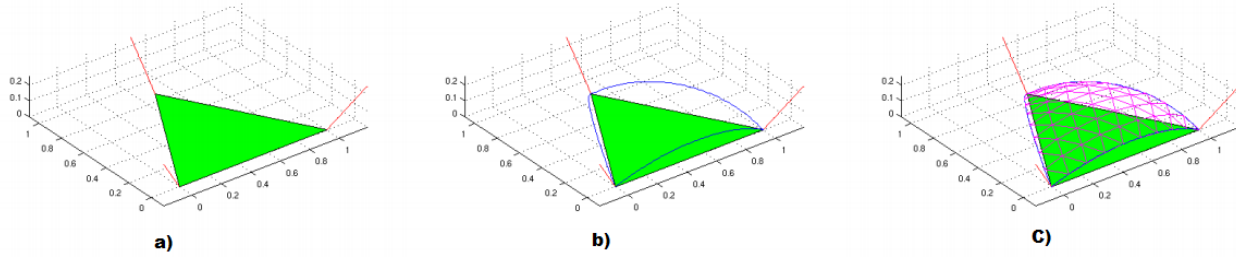
$$n \sum_{k=0}^{n-1} \left(\mathbf{P}_{1,n-1-k,k} - \frac{1}{2} (\mathbf{P}_{0,n-k,k} + \mathbf{P}_{0,n-1-k,k+1}) \right) b_k^{n-1}(w)$$

$$n \sum_{i=0}^{n-1} \left(\mathbf{P}_{i,1,n-1-i} - \frac{1}{2} (\mathbf{P}_{i+1,0,n-1-i} + \mathbf{P}_{i,0,n-i}) \right) b_i^{n-1}(u)$$

$$n \sum_{j=0}^{n-1} \left(\mathbf{P}_{n-1-j,j,1} - \frac{1}{2} (\mathbf{P}_{n-j,j,0} + \mathbf{P}_{n-1-j,j+1,0}) \right) b_j^{n-1}(v).$$
(1)

2.3.4. Lokaliai kvadratinis paviršiaus interpoliavimas

Dabar mes aptarsime problemą interpoliuojant paviršių į viršūnių aibe (v_1, v_2, \dots, v_n) trianguliacijoje $\{T_c\}$. Kiekviename taške v_i mes norime, kad paviršiaus normalė būtų lygiagreti normaliai N_i . Kiekvieno trikampio paviršių iš $\{T_c\}$ mes interpoliuosime ir tokie paviršiaus lopinėliai sudarys visą kompozitinį paviršių.



7 pav. Lokaliios interpoliacijos metodas. a) trikampis ir viršūnių normalės b) Ribinės kreivės statmenos viršūnių normalėms c) paviršiaus lopinėlis kuris atitinka ribines kreives

Pirmiausia mes aprašysime kaip apskaičiuojamos paviršiaus lopinėlių kreivių ribos. Tegu $\mathbf{x}_i, \mathbf{N}_i, i = 0, 1, 2$ būna trikampio viršūnės ir atitinkamos paviršiaus viršūnės normalės. Kiekvienai briaunai mums reikia surasti kreivę $\boldsymbol{\alpha}_i(s)$ kuri tenkintų tokias sąlygas:

$$\begin{cases} \boldsymbol{\alpha}_i(0) = \mathbf{x}_i \\ \boldsymbol{\alpha}_i(1) = \mathbf{x}_{i+1} \\ \boldsymbol{\alpha}'_i(0) \cdot \mathbf{N}_i = 0 \\ \boldsymbol{\alpha}'_i(1) \cdot \mathbf{N}_{i+1} = 0. \end{cases} \quad (2)$$

Pirmos dvi sąlygos skirtos kreivei

$$\boldsymbol{\alpha}_i(s) = \mathbf{x}_i + (\mathbf{d}_i - \mathbf{c}_i)s + \mathbf{c}_i s^2, \quad \mathbf{d}_i = \mathbf{x}_{i+1} - \mathbf{x}_i.$$

Koeficientui \mathbf{c}_i gauname tokias lygtis:

$$\begin{cases} \mathbf{N}_i \cdot (\mathbf{d}_i - \mathbf{c}_i) = 0 \\ \mathbf{N}_{i+1} \cdot (\mathbf{d}_i + \mathbf{c}_i) = 0 \end{cases} \implies \begin{pmatrix} \mathbf{N}_i^T \\ \mathbf{N}_{i+1}^T \end{pmatrix} \mathbf{c}_i = \begin{pmatrix} \mathbf{N}_i \cdot \mathbf{d}_i \\ -\mathbf{N}_{i+1} \cdot \mathbf{d}_i \end{pmatrix}. \quad (3)$$

$$\mathbf{c}_i = \begin{pmatrix} \mathbf{N}_i^T \\ \mathbf{N}_{i+1}^T \end{pmatrix}^+ \begin{pmatrix} \mathbf{N}_i \cdot \mathbf{d}_i \\ -\mathbf{N}_{i+1} \cdot \mathbf{d}_i \end{pmatrix}.$$

Kai nustatome lopinėlio trijų briaunų kreives, mums reikia surasti kvadratinį paviršių kuris sutaptų su tomis kreivėmis. Tegu $\mathbf{y}(s, t)$ būna trikampio paviršiaus lopinėlis aprašytas taip:

$$\mathbf{y}(s, t) = \mathbf{c}_{00} + \mathbf{c}_{10}s + \mathbf{c}_{01}t + \mathbf{c}_{11}st + \mathbf{c}_{20}s^2 + \mathbf{c}_{02}t^2, \quad 0 \leq t \leq s \leq 1. \quad (4)$$

Tuomet trikampio lopinėlis $\mathbf{y}(s, t)$ turėtų tenkintis šias lygybes:

$$\begin{cases} \mathbf{y}(s, 0) = \boldsymbol{\alpha}_0(s) = \mathbf{x}_0 + (\mathbf{d}_0 - \mathbf{c}_0)s + \mathbf{c}_0 s^2 \\ \mathbf{y}(1, t) = \boldsymbol{\alpha}_1(t) = \mathbf{x}_1 + (\mathbf{d}_1 - \mathbf{c}_1)t + \mathbf{c}_1 t^2 \\ \mathbf{y}(s, s) = \boldsymbol{\alpha}_2(1-s) = \mathbf{x}_2 + (\mathbf{d}_2 - \mathbf{c}_2)(1-s) + \mathbf{c}_2(1-s)^2. \end{cases}$$

Kur $\mathbf{c}_i, \mathbf{d}_i, i = 0, 1, 2$ yra koficientai trejoms ribinėms kreivėms. Visi koficientai (4) lygtyje yra nustatomi taip:

$$\begin{cases} c_{00} = \mathbf{x}_0 \\ c_{10} = \mathbf{d}_0 - \mathbf{c}_0 \\ c_{01} = \mathbf{d}_1 + \mathbf{c}_0 - \mathbf{c}_2 \\ c_{11} = \mathbf{c}_2 - \mathbf{c}_0 - \mathbf{c}_1 \\ c_{20} = \mathbf{c}_0 \\ c_{02} = \mathbf{c}_1. \end{cases}$$

2.3.5. Lokalus paviršiaus interpoliavimas naudojant trečiojo laipsnio ribines kreives

2.3.5.1. Paviršiaus lopinėlių ribinių kreivių suradimas

3.3.4. skyriuje mes norėjome surasti kreivę einančia per du duotuosius taškus ir kurių tangentiniai vektoriai yra ortogonalūs dviems duotoms normalėms tuose taškuose. Kreivei tridimensinėje lygtyje (2) suteiktos aštuonios sąlygos. Kvadratinė erdvinė kreivė yra apibrėžiama devyniais parametrais, trimis taškais tridimensinėje erdvėje.

Kubinė kreivė yra apibrėžiama keturiais taškais, 12 parametru. Taigi mums reikia įvesti keletą papildomų sąlygų lygtyje (2) norint apibrėžti kubinę kreivę.

Tegu $\alpha_i(s)$ būna trečiojo laipsnio erdvinė kreivė. Išreiškus $\alpha_i(s)$ kaip Bezje kreivę, turime:

$$\alpha_i(s) = \sum_{k=0}^3 \mathbf{V}_{i,k} b_k^3(s), \quad 0 \leq s \leq 1 \quad (5)$$

Kur b_k^3 yra Bernšteino polinomas laipsniu 3.

Kad unikalčiai apibrėžti ribinę kreivę, įvedami papildomi du apribojimai (2) lygtyje. Tokie apribojimai reikalingi, kad pagrindinė normalė kreives $\alpha_i(s)$ būtų lygiagreti \mathbf{N}_i kai $s = 0$ ir \mathbf{N}_{i+1} kai $s = 1$. Pagrindinė kreivės $\alpha_i(s)$ normalė, pažymima $\hat{\mathbf{N}}_i(s)$ ir ji yra lygiagreti išvestinei kreivės tangentinio vektoriaus.

$$\begin{aligned} \hat{\mathbf{T}}_i(s) &= \frac{\alpha_i'(s)}{\|\alpha_i'(s)\|} = \frac{\alpha_i'(s)}{(\alpha_i'(s) \cdot \alpha_i'(s))^{1/2}}, \\ \hat{\mathbf{N}}_i(s) &= k(s) \frac{d\hat{\mathbf{T}}_i(s)}{ds} = \\ &= k(s) \frac{\alpha_i''(\alpha_i' \cdot \alpha_i')^{1/2} - \alpha_i'(\alpha_i' \cdot \alpha_i')^{-1/2}(\alpha_i' \cdot \alpha_i'')}{(\alpha_i' \cdot \alpha_i')}, \end{aligned}$$

Kur $k(s)$ yra normalizuotas skaliaras. Padauginus išraišką iš $(\alpha_i' \cdot \alpha_i')^{1/2}$, mes gauname jog pagrindinė normalė yra lygiagreti

$$\alpha_i''(\alpha_i' \cdot \alpha_i') - \alpha_i'(\alpha_i' \cdot \alpha_i'').$$

Sekanti sąlyga būtų tokia:

$$\begin{aligned} \frac{d\|\alpha_i'(s)\|}{ds} &= \frac{d(\alpha_i' \cdot \alpha_i')^{1/2}}{ds} = \frac{\alpha_i' \cdot \alpha_i''}{\|\alpha_i'\|} = 0 \\ \implies \alpha_i' \cdot \alpha_i'' &= 0. \end{aligned}$$

Apibendrinus, mes naudosime šias sąlygas kartu su pirmomis dviem sąlygomis lygtyje (2), kad apibrėžti paviršiaus lopinėlio kubinę kreivės ribą:

$$\begin{cases} \alpha_i'(0) \cdot \alpha_i''(0) = 0 \\ \alpha_i'(1) \cdot \alpha_i''(1) = 0 \\ \alpha_i''(0)(\alpha_i'(0) \cdot \alpha_i'(0)) = \rho_i \mathbf{N}_i \\ \alpha_i''(1)(\alpha_i'(1) \cdot \alpha_i'(1)) = \sigma_i \mathbf{N}_{i+1} \end{cases}$$

Išsprendę šitas lygtis pagal (5), gauname

$$\begin{aligned} \mathbf{V}_{i,0} &= \mathbf{x}_i \\ \mathbf{V}_{i,1} &= \mathbf{x}_i + \frac{6\mathbf{D}_i - 2\rho_i \mathbf{N}_i + \sigma_i \mathbf{N}_{i+1}}{18} \\ \mathbf{V}_{i,2} &= \mathbf{x}_{i+1} - \frac{6\mathbf{D}_i + \rho_i \mathbf{N}_i - 2\sigma_i \mathbf{N}_{i+1}}{18} \\ \mathbf{V}_{i,3} &= \mathbf{x}_{i+1}. \end{aligned} \tag{6}$$

Su

$$\begin{aligned} \rho_i &= \frac{6(2\mathbf{N}_i \cdot \mathbf{D}_i + (\mathbf{N}_i \cdot \mathbf{N}_{i+1})\mathbf{N}_{i+1} \cdot \mathbf{D}_i)}{4 - (\mathbf{N}_i \cdot \mathbf{N}_{i+1})^2} \\ \sigma_i &= \frac{6(2\mathbf{N}_{i+1} \cdot \mathbf{D}_i + (\mathbf{N}_i \cdot \mathbf{N}_{i+1})\mathbf{N}_i \cdot \mathbf{D}_i)}{4 - (\mathbf{N}_i \cdot \mathbf{N}_{i+1})^2} \\ \mathbf{D}_i &= \frac{\mathbf{x}_{i+1} - \mathbf{x}_i}{\|\mathbf{x}_{i+1} - \mathbf{x}_i\|}. \end{aligned}$$

2.3.5.2. Ribų tangentinės plokštumos

Kiekviename taške išilgai ribos, mums reikia surasti du vektorius, kurie apimtų tinkamą paviršiaus tangentinę plokštumą tame taške. Kaip jau buvo minėta anksčiau mes norime, kad šitie vektoriai būtų apibrėžti ribinėmis kreivėmis. Vienas akivaizdus pasirinkimas būtų paimti pirmą vektorių kaip kreivės tangentą, $\mathbf{v}_{i,1} = \alpha_i'(s)$.

$$\mathbf{v}_{i,1}(s) = 3 \sum_{k=0}^2 (\mathbf{V}_{i,k+1} - \mathbf{V}_{i,k}) b_k^2(s) = 3 \sum_{k=0}^2 \mathbf{W}_{i,k} b_k^2(s), \quad 0 \leq s \leq 1$$

Pasirenkame antrąjį vektorių

$$\mathbf{v}_{i,2}(s) = \sum_{k=0}^2 \mathbf{A}_{i,k} b_k^2(s), \quad 0 \leq s \leq 1$$

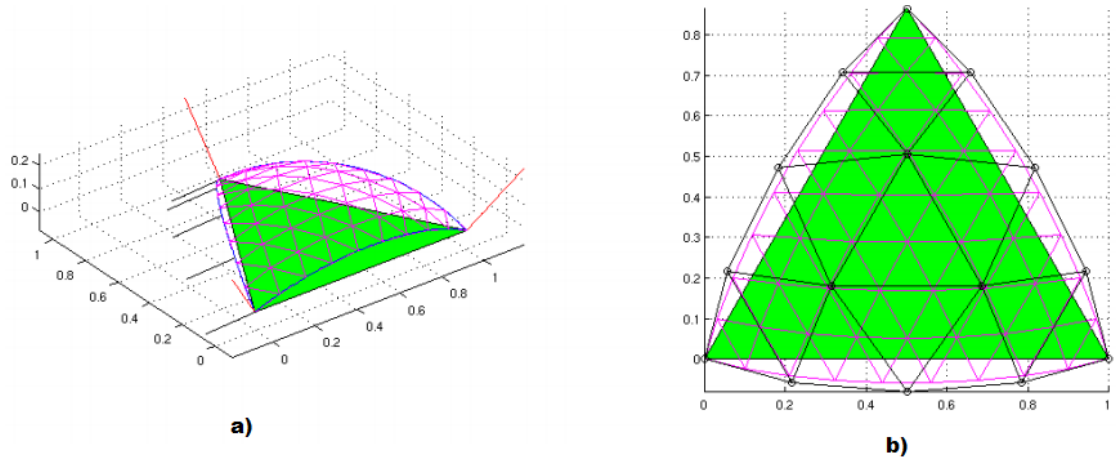
Kur kontroliniai taškai $\mathbf{A}_{i,k}$ yra duoti tokie

$$\begin{aligned} \mathbf{A}_{i,0} &= \frac{\mathbf{N}_i \times \boldsymbol{\alpha}'_i(0)}{\|\boldsymbol{\alpha}'_i(0)\|} \\ \mathbf{A}_{i,2} &= \frac{\mathbf{N}_{i+1} \times \boldsymbol{\alpha}'_i(1)}{\|\boldsymbol{\alpha}'_i(1)\|} \\ \mathbf{A}_{i,1} &= \frac{(\mathbf{A}_{i,0} + \mathbf{A}_{i,2})}{\|\mathbf{A}_{i,0} + \mathbf{A}_{i,2}\|}. \end{aligned}$$

2.3.5.3. Paviršiaus lopinėlio sudarymas

Ketvirto laipsnio Bezje lopinėlis yra apibrėžiamas 15 kontrolinių taškų. Lygtis tokiam lopinėliui būtų

$$\mathbf{S}(u, v, w) = \sum_{i+j+k=4} \mathbf{P}_{i,j,k} b_{i,j,k}^4(u, v, w), \quad i, j, k \geq 0. \quad (7)$$



8 pav. a) vektorius $\mathbf{v}_{i,2}$ nubrėžtas išilgai ribos b) ketvirto laipsnio trikampio lopinėlio kontroliniai taškai

Išilgai kiekvienos briaunos mes turime 5 kontrolinius taškus. Taškai yra surandami pakeliant laipsniu ribines kreives iki 4 laipsnio polinomo. Tai užtikrina, kad paviršius tęstųsi išilgai lopinėlių paviršiaus, kadangi du gretimi lopinėliai dalinasi ta pačia ribine kreive. Mes sukursime paviršių G^1 pakoreguojant 3 vidinių kontrolinių taškų pozicijas, $\mathbf{P}_{2,1,1}$, $\mathbf{P}_{1,2,1}$ ir $\mathbf{P}_{1,1,2}$. Kryžminės ribos išvestinė yra duota lygtyje (1). Tegu $F_i(s)$ būna kryžminės ribos išvestinė išilgai ribos i , $i = 0, 1, 2$. Ketvirtojo laipsnio lopinėliui F_i , bus kubinė Bezje kreivė.

$$\mathbf{F}_i(s) = \sum_{k=0}^3 \mathbf{D}_{i,k} b_k^3(s)$$

Kontroliniai taškai $D_{i,k}$ gali būti surasti iš lygties (1)

$$\begin{aligned} D_{0,k} &= 4 \left(P_{1,3-k,k} - \frac{1}{2} (P_{0,4-k,k} + P_{0,3-k,k+1}) \right) \\ D_{1,k} &= 4 \left(P_{k,1,3-k} - \frac{1}{2} (P_{k,0,4-k} + P_{k+1,0,3-k}) \right) \\ D_{2,k} &= 4 \left(P_{3-k,k,1} - \frac{1}{2} (P_{4-k,k,0} + P_{3-k,k+1,0}) \right) \end{aligned} \quad (8)$$

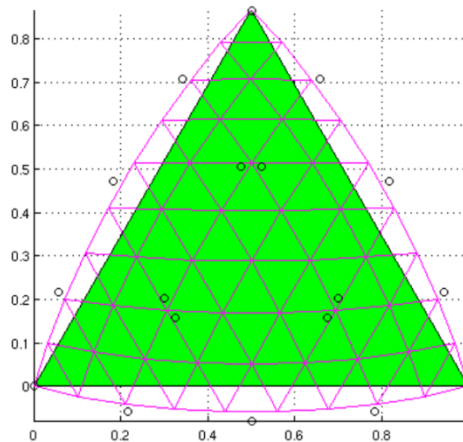
Dabar mums reikia kad $F_i(s)$ apgaubtų plokštumą aprėpiant $v_{i,1}(s)$ ir $v_{i,2}(s)$

$$F_i(s) = f_i(s)v_{i,1}(s) + g_i(s)v_{i,2}(s). \quad (9)$$

$v_{i,1}(s)$ ir $v_{i,2}(s)$ yra kvadratinės Bezje kreivės, taigi pasirinkus $f_i(s)$ ir $g_i(s)$ kaip pirmojo laipsnio Bezje kreives, mes gauname kubinę Bezje kreivę kiekvienoje pusėje lygtyje (9). Turint omenyje, kad mes žinome visus kontrolinius taškus išilgai lopinėlio ribos, mes galime išspręsti šią lygtį kiekvienam i , kad gauti vidinių kontrolinių taškų pozicijas. Pagrindinė problema čia, yra ta, kad kiekvienas vidinis kontrolinis taškas, duos dvi atskiras pozicijas, po vieną kiekvienam kaimynui. Išsprendžiame tai, pasinaudoję kombinacija tų dviejų pozicijų, vietoj to kad naudotume įprastinius kontrolinius taškus Bezje lopinėlyje. Tegu $G_{i,k}$, $k = 0, 1$ būna du kontroliniai taškai susieti su kiekviena briauna $i = 0, 1, 2$, kuriuos gauname iš lygties (9). Tuomet nustatome vidinius kontrolinius taškus

$$\begin{aligned} P_{1,1,2} &= \frac{1}{u+v} (uG_{1,0} + vG_{0,1}) \\ P_{2,1,1} &= \frac{1}{v+w} (vG_{2,0} + wG_{1,1}) \\ P_{1,2,1} &= \frac{1}{w+u} (wG_{0,0} + uG_{2,1}). \end{aligned}$$

Vietoj Bezje lopinėlio, dabar mes turime Gregorijaus (angl. Gregory) lopinėlių.



9 pav. Kiekvienam vidiniam kontroliniam taškui duotos dvi pozicijos

2.4. Baigtinių elementų metodas

Baigtinių elementų metodas (BEM) taikomas įvairius fizikinius reiškinius ir procesus aprašančioms diferencialinėms lygtims dalinėmis išvestinėmis spręsti. Skaičiavimai baigtinių elementų metodu taikomi šiais etapais:

- bet kuri sudėtingos geometrinės formos sritis pavaizduojama baigtiniu nesudėtingos formos elementų skaičiumi (tarkim trikampaiais, tetraedrais ir pan.);
- kiekvieno elemento srityje diferencialinė lygtis dalinėmis išvestinėmis apytiksliai pakeičiama algebrinių lygčių sistema;
- elementų algebrinės lygtys jungiamos į bendrą vienareikšmiškai išsprendžiamą algebrinių lygčių sistemą.

Savo darbe naudosime išreikštinį (angl. explicit) analizės būdą. Taikant išreikštinį būdą, apskaičiuojami sistemos pagreičiai ir pagal juos prognozuojama, kaip konstrukcijos mazgai pasislinks labai trumpo laiko intervalo bėgyje. Konstrukcijos mazgams pasislinkus, pagreičiai apskaičiuojami iš naujo, įvertinant pakitusius konstrukcijos poslinkius ir greičius, prognozuojamos poslinkių reikšmės sekančio trumpo laiko intervalo bėgyje, ir procesas kartojamas.

2.4.1. Netiesinis dinamikos uždavinys. Bendroji formuluotė.

Kintant pridėtoms apkrovoms arba judant kūnams atsiranda taip vadinami dinaminiai efektai, kuriuos sukelia didesni greičiai ar jų pokyčiai. Dėl šių poveikių susidaro papildomos nuomasių išsidėstymo ir konstrukcijos savybių priklausančios inercijos ir slopinimo jėgos. Kūnuose su daugeliu laisvės laipsniu jų poveikis gana sudėtingas ir jo negalima išreikšti paprastu dinaminio proporcingumo koeficientu. Išcentrinės jėgos, smūginės apkrovos, virpesiai, rezonansas, bangų sklidimas yra būdingi dinaminių reiškinių pavyzdžiai. Dinaminis reiškinius nagrinėja mechanikos šaka dinamika, kuriai priskiriamos įvairios teorinės disciplinos, tokios kaip virpesių ar bangų teorijos, ir taikomosios disciplinos kaip statinių, mašinų ar mechanizmų teorijos.

Trumpai aptarsime bendrąjį netiesiškai deformuojamo kūno atvejį. Kūno būvį aprašo netiesinė dinaminė pusiausvyros lygtis, išreikšta poslinkiais. Netiesinėse sistemose būvis yra sudėtingas laiko bėgyje kintantis procesas, kur būvio kintamieji yra laiko funkcijos.

Užrašant būvio lygtį diskretinei struktūrai, kurios dalinis atvejis yra iš baigtinių elementų sudaryta struktūra, struktūros mazginių poslinkių vektorius $\{U\}$ tampa pagrindiniu modelio nežinomuoju $\{U\} \equiv \{U(t)\}$.

Nesigilindami į laiko diskretizacijos ypatumus pagrindinius nežinomuosius apibrėžiame formalia priklausomybe.

Netiesiškai deformuojamos sistemos dinaminės pusiausvyros lygtį pateiksime be išvedimo, užrašydami ją tokia lygčių sistema arba matricine lygtimi:

$$\begin{aligned} & [M(\{\ddot{U}\}, \{\dot{U}\}, \{U\}, t)] \{\ddot{U}(t)\} - [C(\{\ddot{U}\}, \{\dot{U}\}, \{U\})] \{\dot{U}(t)\} + \\ & + [K(\{\ddot{U}\}, \{\dot{U}\}, \{U\}, t)] \{U(t)\} = \{F(\{\ddot{U}\}, \{\dot{U}\}, \{U\}, t)\} \end{aligned} \quad (10)$$

Pirmasis narys išreiškia inercijos jėgas, antrasis - slopinimo jėgas, trečiasis - sistemos vidines jėgas, o laisvasis narys - išorines jėgas. Ši lygtis yra netiesinio dinamikos uždavinio matematinis modelis. Kaip galima pastebėti visi į modelį įeinantys dydžiai visų pirma yra laiko t funkcijos, o taip pat priklauso nuo kūno būvio kintamųjų bei jų išvestinių laiko atžvilgiu reikšmių. Čia $\{U(t)\}$ yra nežinomas diskretinės sistemos poslinkių vektorius, $\{\ddot{U}(t)\} = \left\{ \frac{d^2 U}{dt^2} \right\}$ yra

pagreičiai, $\{\dot{U}(t)\} = \left\{ \frac{dU}{dt} \right\}$ - greičiai. Duotus dydžius sudaro: $[M]$ masių matrica, $[C]$ - slopinimo matrica, $[K]$ - standumo matrica, $\{F\}$ - išorinių apkrovų vektorius, kurių reikšmės gali kisti deformavimo eigoje.

Išraiška (10) yra pakankamai bendra ir gali praktiškai įvertinti visus deformuojamame kūne pasitaikančius efektus, įskaitant dinamiką, temperatūrą, valkšnumą, plastiškumą, geometrinį netiesiškumą, struktūrinius pakitimus ir pan. Priimant įvairius supaprastinimus, iš šio modelio galima sudaryti daugybę paprastesnių modelių aprašančių konkrečius efektus.

Bendroji išraiška (10) turi algebrinę-diferencialinę formą ir nepriklauso nuo konkretaus diskretizacijos metodo. Nagrinėjant nežinomų funkcijų kitimą kūno tūryje diskretinių metodų pagalba, diferencialinės kūno priklausomybės yra išreiškiamos algebriniais operatoriais, tuo tarpu priklausomybė nuo laiko yra palikta diferencialinėje formoje. Koeficientų matricių $[M]$, $[C]$ ir $[K]$ bei laisvųjų narių vektoriaus $\{F\}$ sudarymas baigtinių elementų metodu yra analogiškas tiesiniams uždaviniams. Šios matricos pirmiausia sudaromos atskiriems baigtiniams elementams, o visos sistemos matricos sudaromos naudojant standartinę elementų surinkimo procedūrą. Išimtis yra slopinimo matrica $[C]$, kuri dažnai sudaroma kaip $[M]$ ir $[K]$ kombinacija.

Diskretizuojant laiką, modelis (10) yra daugiau simbolinis ir nekalba apie laiko diskretizaciją, kaip jau buvo aptarta anksčiau. Diskretizuojant laiką bet koks procesas yra aprašomas kaip sistemos (kūno) būsenų atskirais laiko momentais seka naudojant anksčiau aptartas netiesinės mechanikos sąvokas ir metodus, susijusius su dydžių nagrinėjimu konkrečiuose konfigūracijose. To pasekoje lygtis (10) gali turėti keletą skirtingų realizavimo modelių priklausančių nuo įtempimų ir deformacijų matų (konkrečių įtempimų ir deformacijų tenzorių) bei diskretizacijos laike būdų. Reiškia formali priklausomybė $U(t)$ irgi susiejama su konkrečiomis kūno konfigūracijomis. Tas pats pasakytina apie greičius $\dot{U}(t)$ ir pagreičius $\ddot{U}(t)$, taip pat lygties koeficientus $[M(t)]$, $[C(t)]$, $[K(t)]$ ir laisvuosius narius $F(t)$. Apie tokius dalykus užsiminsime nagrinėdami konkretesnius uždavinius, o kol kas apsiribojama pačiais bendriausiais teiginiais.

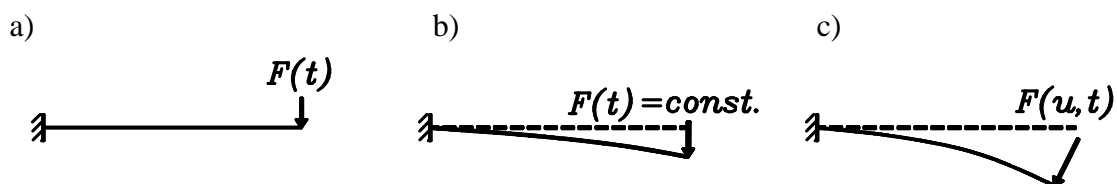
Atmetus galimus struktūrinius deformuojamos sistemos pakitimus, praktikoje dažniausia naudojama šiek tiek suprastinta modelio (10) išraiška:

$$[M]\{\ddot{U}(t)\} + [C]\{\dot{U}(t)\} + [K(\{U\}, t)]\{U(t)\} = \{F(\{\dot{U}\}, \{U\}, t)\}. \quad (11)$$

Kaip matote čia $[M]$ ir $[C]$ nepriklauso nuo būvio kintamųjų, o $[K]$ priklauso tik nuo poslinkių ir laiko.

2.4.2. Apkrovos priklausomybė nuo būvio kintamųjų.

Tiesinėje mechanikoje neiškyla apkrovų apibrėžimo problemos. Apkrovoms galioja prielaida, kad jos gali priklausyti tik nuo laiko ir visiškai nepriklauso nuo sistemos padėties (būvio kintamųjų reikšmių). To negalima pasakyti apie netiesines sistemas. Apkrovos padėties ir dydžio nustatymas judančiose ir besideformuojančiose sistemose reikalauja papildomų prielaidų ir hipotezių. Panagrinėkim paprasčiausią pavyzdį (10 pav.). Tiesinėse sistemose įprasta nagrinėti konservatyvią apkrovą (10 b pav.), kuri sistemai deformuojantis nepriklauso nuo sistemos būvio. Antruoju atveju (10 c pav.) apkrova juda ir keičiasi kartu su deformuojama sistema, o toks apkrovimas yra vadinamas nekonservatyviu. Jeigu sistema juda, gali atsirasti papildomų jėgų, priklausančių nuo judėjimo greičio. Dažnai tokios jėgos atsiranda dėl Koriolio pagreičio.



10 pav. Konservatyvios (b) ir nekonservatyvios (c) apkrovos iliustracija

Kitu nekonservatyvios apkrovos pavyzdžiu yra skysčių ir dujų slėgis į kietus kūnus, kuris priklauso nuo kūno būvio. Apibendrintai visą sistemą veikiančią apkrovą galima suskaidyti į atskiras dedamąsias:

$$\{F\} = \{F_p(t)\} + \{F_u(U, t)\} + \{F_v(\dot{U}, t)\}. \quad (12)$$

Čia $\{F_p\}$ - nuo kūno būvio nepriklausanti apkrova, $\{F_u\}$ yra nuo poslinkių, o $\{F_v\}$ - nuo greičių priklausančios apkrovos. Nuo poslinkių priklausanti apkrova yra susijusi su konstrukcijos standumu konkrečiu laiko momentu ir gali būti išreikšta priklausomybe:

$$\{F_u(\{U\}, t)\} = [K_F(\{U\}, t)]\{U(t)\}. \quad (13)$$

Ją galima traktuoti kaip papildomą vidinę sistemos jėgą.

Analogiškai galima išreikšti ir nuo greičio priklausančią jėgą ir traktuoti ją kaip papildomą slopinimo jėgą.

$$\{F_v(\{\dot{U}\}, t)\} = [C_F(\{\dot{U}\}, t)]\{\dot{U}(t)\}. \quad (14)$$

Įstačius jėgas į (11) galima gauti tokią būvio lygtį.

$$[M]\{\ddot{U}(t)\} + ([C] - [C_F(\{\dot{U}\}, t)])\{\dot{U}(t)\} + ([K(\{U\}, t)] - [K_F(\{U\}, t)])\{U(t)\} = \{F_p(t)\}. \quad (15)$$

Nenagrinėdami detalios apkrovų klasifikacijos plačiau, aptarsime *konservatyvumo* sąvoką. Jeigu apkrovų darbas priklauso tik nuo sistemos pradinio ir galutinio būvio, apkrovos laikomos *konservatyviomis*, o jei darbas priklauso nuo apkrovimo kelio - *nekonservatyviomis*. Esant konservatyvioms apkrovoms, visos lygties (11) matricos yra simetrinės, o esant nekonservatyviomis $[C_F]$ ir $[K_F]$ - nesimetrinės. Tuo pačiu nesimetrinės yra ir suminės matricos $[C]$ ir $[K]$. Apkrovos konservatyvumą taip pat galima susieti su tam tikromis energetinėmis savybėmis.

Esant konservatyvioms apkrovoms, (neįvertinant apkrovų priklausomybės nuo būvio kintamųjų), vietoje išraiškų (10) ar (11) dažniausia naudojama tokia dinaminės pusiausvyros lygties išraiška:

$$[M]\{\ddot{U}(t)\} + [C]\{\dot{U}(t)\} + [K(\{U\}, t)]\{U(t)\} = \{F(t)\}. \quad (16)$$

Atmetus inercijos ir slopinimo jėgos lygtis (16) virsta netiesine statikos lygtimi:

$$[K(\{U\}, t)]\{U(t)\} = \{F(t)\}$$

Nesunku pastebėti, kad pirmuosiuose skyriuose aptartas tiesinės dinamikos uždavinys

$$[M]\{\ddot{U}(t)\} + [C]\{U(t)\} + [K]\{U(t)\} = \{F(t)\}$$

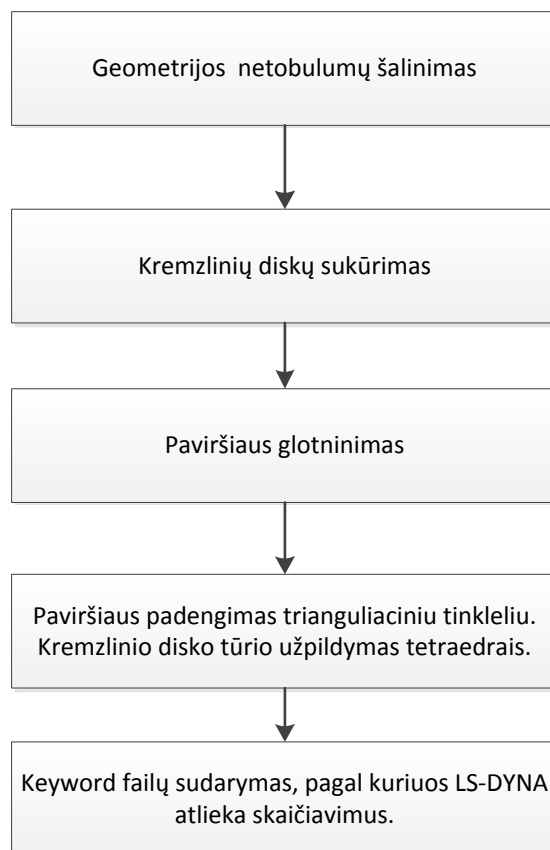
yra dalinis netiesinio uždavinio (10) atvejis, o tiesinis statikos uždavinys

$$[K]\{U(t)\} = \{F(t)\} \text{ yra dalinis uždavinio (16) atvejis.}$$

3. PROJEK TINĖ DALIS

3.1. Projektavimo algoritmo struktūra

11 paveiksle yra pateiktas modelio sukūrimo veiksmų seka.



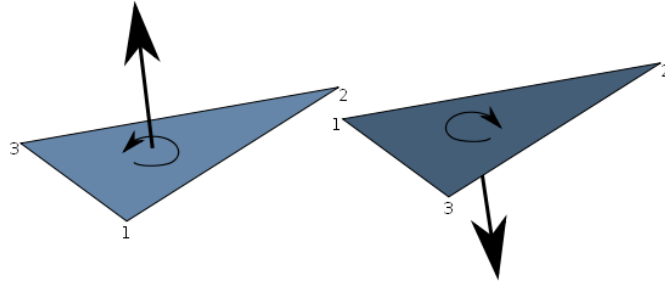
11 pav. Projektavimo algoritmas.

3.2. Geometrijos netobulumų šalinimas.

Dažnai pasitaiko, jog STL duomenyse kurie aprašo geometrija atsiranda klaidų. Dėl to geometrija atvaizduojama netinkamai. Geometrijos netobulumai gali įnešti klaidų skaičiavimuose baigtinų elementų metodu, taip pat apsunkina paviršiaus glotninimo procedūrą. Šiuos netobulumus pašalinsime pasitelkę netFabb studio programinę įrangą. Žemiau pateiktas sąrašas su pašalintomis problemomis:

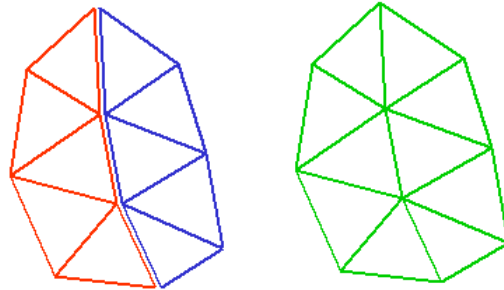
- **Trikampių orientacijos klaida.** Geometrinio objekto orientacija yra apsprendžiama visų paviršiaus trikampių orientacija. Viršūnių seka apibūdinanti trikampi, nulemia trikampio orientaciją pagal dešinės rankos taisyklę. Jeigu visų trikampių orientacija vienoda, tuomet paviršius padengtas tinkamai, bet jeigu nors

vienas trikampis su bloga orientacija, tuomet reikia jį apversti. Paprastai tokia klaida gali įnešti klaidų baigtinių elementų skaičiavimuose.



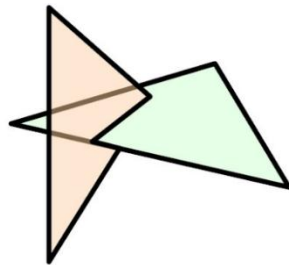
12 pav. Dešinės rankos taisyklė trikampiams

- **Nesujungti trikampiai.** Tarp kaimyninių trikampių atsiradęs tarpas. Du kaimyniniai trikampiai privalo dalintis ta pačia kraštine.



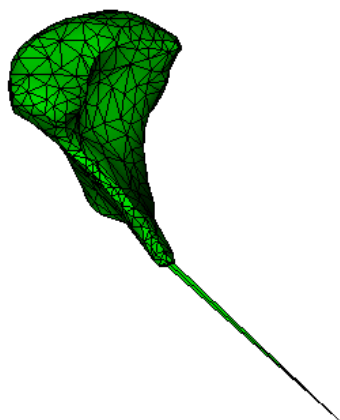
13 pav. Nesujungti trikampiai

- **Trikampių susikirtimas**



14 pav. Susikirtę trikampiai

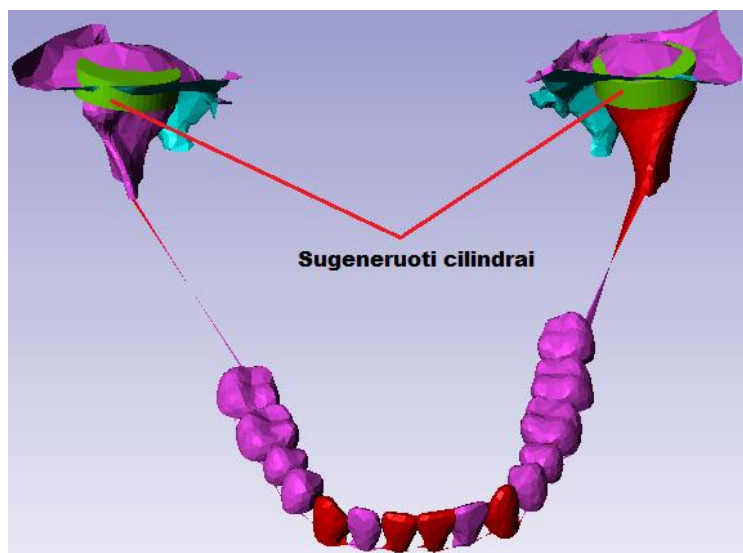
- **Smalių išsikišimų pašalinimas.** Tai nėra geometrijos klaida, tačiau pašalinus tokius trikampius lengviau glotninamas objekto paviršius. Reikia pridurti, jog pašalinti tokie trikampiai nedaro įtakos baigtinių elementų skaičiavimuose, kadangi juose nevyksta kontakto jėgos.



15 pav. Smailus išsikišimas prie sąnario

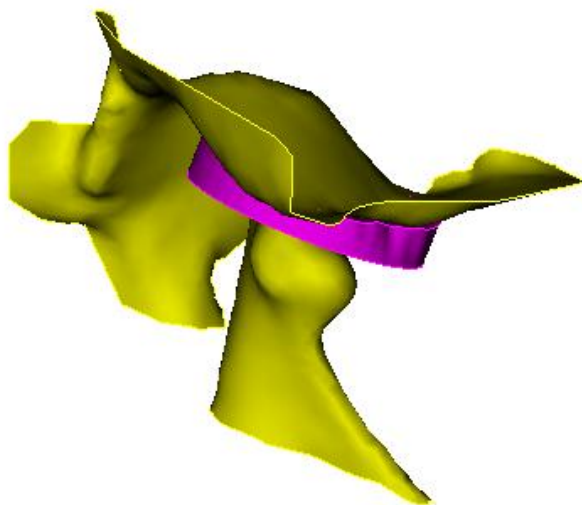
3.3. Kremzlinių diskų sukūrimas

Kompiuterinis tomografas skenuoja kietus audinius. Mūsų atveju pradiniuose nuskenuotose duomenyse tai būtų apatinis ir viršutinis žandikauliai ir smilkinkaulio apatinio žandikaulio duobutės. Kadangi visa darbo esmė išmatuoti kremzlinių sąnarių diskų slėgio jėgas, reikėjo surasti būdą kaip dirbtinai sukurti kremzlinius diskus. Buvo nuspręsta sukurti cilindro formos geometrinius primityvus ir atlikti paviršiaus Būlio operacijas. Šias procedūras buvo atliktos pasitelkiant ScanIP +CAD ir Rhinoceros programinę įrangą. ScanIP +CAD aplinkoje labai patogiu susigeneruoti cilindro formos primityvą ir interaktyviai keisti dydį, pozicionuoti erdvėje. Cilindro skersmuo buvo apytiksliai parinktas toks, kad apgaubtų apatinio žandikaulio sąnarį.



16 pav. Sukurti cilindrai ir įstatyti tarp apatinio žandikaulio sąnarių ir žandikaulinių duobučių

Matome, jog cilindrai yra padėti tarp apatinio žandikaulio sąnarių ir smilkinkaulio apatinio žandikaulio duobučių. Esama cilindrų forma neatitinka realių kremzlinių diskų, nes cilindrų geometrija yra įsiskverbus į sąnario ir duobučių kaulines struktūras. Kad panaikinti cilindrų įsiskverbimą, buvo panaudotos Būlio paviršiaus atimties operacijos su Rhinoceros programine įranga. Mums reikalinga toks kremzlinio disko paviršius kuris gražiai apgaubtų sąnario paviršių ir įsiremtų į duobutės paviršių, todėl iš cilindrų buvo atimtas sąnario ir duobučių paviršių reljefai, taip suformuojant apytiksliai realybeje atitinkančius kremzlinius diskus.

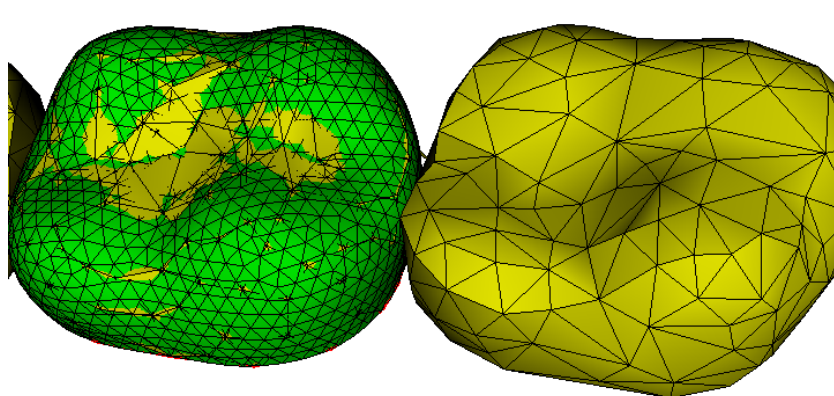


17 pav. Sukurtas kremzlinis sąnarinis diskas

3.4. Paviršiaus glotninimas

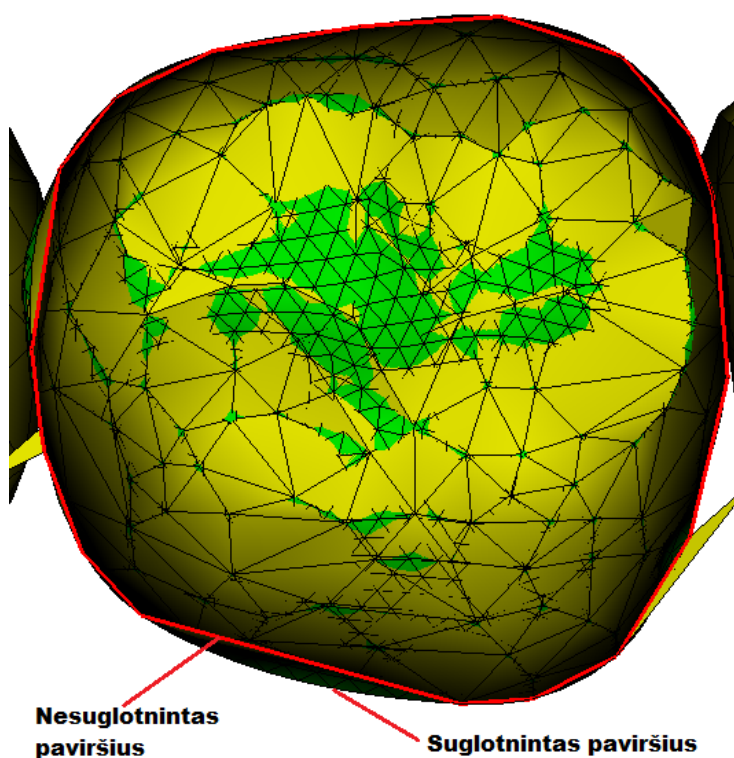
Esamas paviršius atvaizduoja geometrija ganėtinai „šurkščiai“, todėl buvo nuspręsta suglotninti paviršių. Paviršiaus lygumas labiausiai reikalingas tose srityse, kur vykstant sukandimui kontaktuoja paviršiai. Realiai dantų paviršiai daugelyje vietų yra glotnūs, o esama geometrija dantys atvaizduoja ganėtinai stambiais trikampaiais, todėl įvairūs išlenkimai būna aštrūs. Jei toks aštrus paviršius kontaktuoja su kitu paviršių, dantys prasčiau „sukimba“ (nepilnas sukandimas) ir iškreipia skaičiavimo rezultatus. Pagal fizikos apibrėžimus, kuo kūno paviršiaus plotas mažesnis, tuo jis sukelia didesnę slėgį į kontaktuojantį paviršių. Sakykime kremzlinio disko geometrijoje būtų smailus išsikišimas, nors realiai kremzliniai diskai turi glotnų paviršių. Kadangi smailumas turi labai mažą paviršiaus plotą, tai vykstant kontaktui sąnarys į kremzlę, bus iššauktos didelės slėgio jėgos smailės susidūrimo srityje. Kad to išvengti reikia perteikti geometrija kiek galima panašesne į realaus pasaulio duomenys.

Paviršiaus glotninimui panaudosime Cubit programinę įrangą. Cubit yra įgyvendinę paviršiaus glotninimo algoritmą aprašytą teorinėje 3.3.5. dalyje. Žemiau pateiktas danties paviršiaus suglotninimo pavyzdys:



18 pav. Suglotnintas žalias paviršius „užvilktas“ ant nesuglotninto paviršiaus

Pavyzdyje žalias danties paviršius reprezentuoja suglotnintą paviršių, o geltonas originalų paviršių. Kad būtų lengviau išvelgti skirtumus suglotnintas paviršius yra „užvilktas“ ant nesuglotninto paviršiaus.



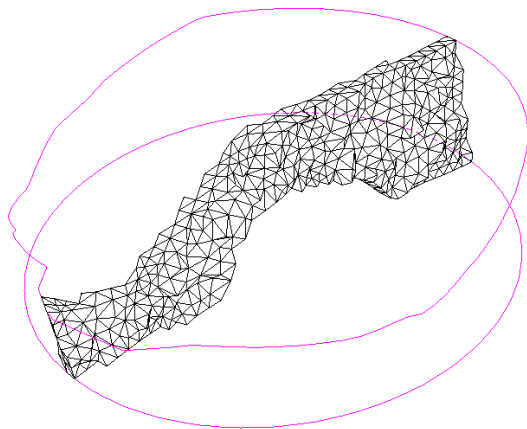
19 pav. Danties pjūvis iš apačios

Pateikiamas danties pjūvis iš apačios, kuriame aiškiai matosi skirtumas: suglotnintas paviršius apgaubia „laužyta“ (raudona kreivė) paviršių.

3.5. Paviršiaus padengimas trianguliaciniu tinkleliu ir kremzlinių diskų tūrio užpildymas tetraedrais

Paviršiaus suskaidymas į smulkius trikampus yra vienas iš paruošiamųjų darbų norint pradėti baigtinių elementų analizę. Šiam darbui vėlgi buvo panaudota Cubit programinė įranga. Visi paviršiai, išskyrus kremzlinius diskus buvo aptraukti tinkleliu kurį sudaro 30815 trikampiai.

Kadangi kremzlinis diskas bus deformuojamas slėgio, šiam objektui nepakanka padengti paviršių tik trikampaiais elementais. Objektai, kurių padengtas tik paviršius yra laikomi standžiais kūnais (vykstant sukandimui nesideformuoja). Tuo tarpu kad atvaizduoti kremzlinio disko deformacijas erdvėje, būtina jo tūrį užpildyti. Tūrį buvo pasirinkta užpildyti 79365 tetraedrinėmis figūromis.



20 pav. Kremzlės tūris pildomas tetraedrais.

Bendras sugeneruotų elementų kiekis sudaro 110180. Kontakto vietose reikalingas tankus elementų kiekis, tačiau per daug sutankinus pastebimai prailgėja skaičiavimo trukmė, todėl buvo pasirinktas kompromisinis variantas.

3.6. Keyword failų sudarymas

Baigtinių elementų analizę atliksime su LS-DYNA programine įranga. LS-DYNA analizė pradedama paleidus keyword failus. Per keyword failus yra organizuojamas visas darbas LS-DYNA aplinkoje. Buvo sukurti 4 keyword failai. Kiekvieno paskirtį aptarsime trumpai.

- **elements_nodes.k**

Šis failas aprašo modeliuojamos geometrijos elementus (trikampus, tetraedrus) ir elementų mazgų koordinates. Šis failas buvo sudarytas eksportuojant Cubit aplinkoje sukurtą geometriją į keyword failą.

Duomenų apdorojimas keyword failuose vyksta per identifikatorius ir identifikatoriui priskirtas reikšmes. Pavyzdžiui, `elements_nodes.k` failo pradžioje galime pamatyti tokias eilutes:

```
*KEYWORD
*NODE
$   nid           x           y           z
 21647      0.0117421   0.00368354   0.0508787
 21648      0.0117023   0.00391845   0.0503659
 21649      0.01137     0.00423507   0.0505444
```

Keyword failai pradedami `*KEYWORD` ir baigiami `*END` žodžiais. Toliau `*NODE` nurodo kad bus aprašomi elementų mazgai. Kiekvienas mazgas turi unikalų identifikatorių `nid` su jam priskirta reikšme. Apsirašius visus geometrijos mazgus, toliau aprašomos modelio dalys ir toms dalims priklausantys elementai suformuoti iš mazgų aprašytų aukščiau. Dalys savo ruožtu gali būti arba tūrinės arba kevalinės (paviršinės). Tarkim reikia aprašyti tūrinę dalį (kremzlinius diskus). Tuomet:

```
$   eid   pid     n1     n2     n3     n4     n5     n6     n7     n8
*ELEMENT_SOLID
   1     1  21647  21648  21649  21650  21650  21650  21650  21650
   2     1  21647  21651  21649  21652  21652  21652  21652  21652
```

`*ELEMENT_SOLID` pasako, kad tai tūrinis elementas. `pid` – dalies (mūsų atveju kremzlės) identifikacinis numeris geometrijoje, `eid` – elemento identifikacinis numeris, `n1...n8` mazgu identifikaciniai numeriai atitinkantys aukščiau aprašytus `nid`.

Tolesnis duomenų srautų „vaikščiojimas“ vyksta panašiu principu. Aprašytoms geometrijos dalims pagal ID priskiriamos medžiagų savybės, jėgos ir t.t.

- **mandibula_nodes.k**

Šiame faile yra sukurti 8 mazgai ir tiems mazgams priskirtos jėgos, kurių vektoriai statmenai nukreipti į viršutinį žandikaulį. Mazgų pozicijos atitinka apatinio žandikaulio raumenų vietas.

- **materials.k**

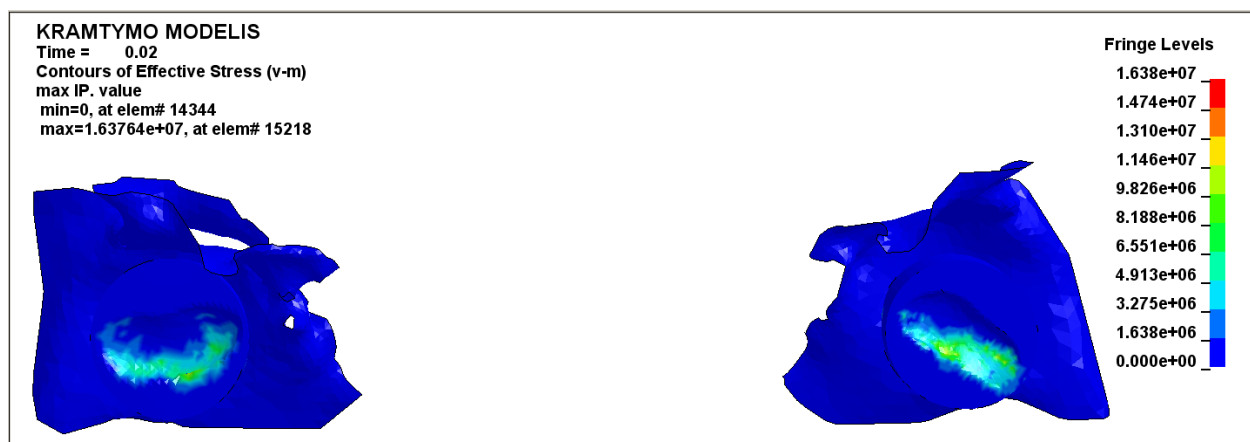
Aprašomos medžiagų savybės. Kremzliniams diskams priskirtos elastiško kūno savybės, o visos kitos dalys laikomos standžiais kūnais.

- **analize.k**

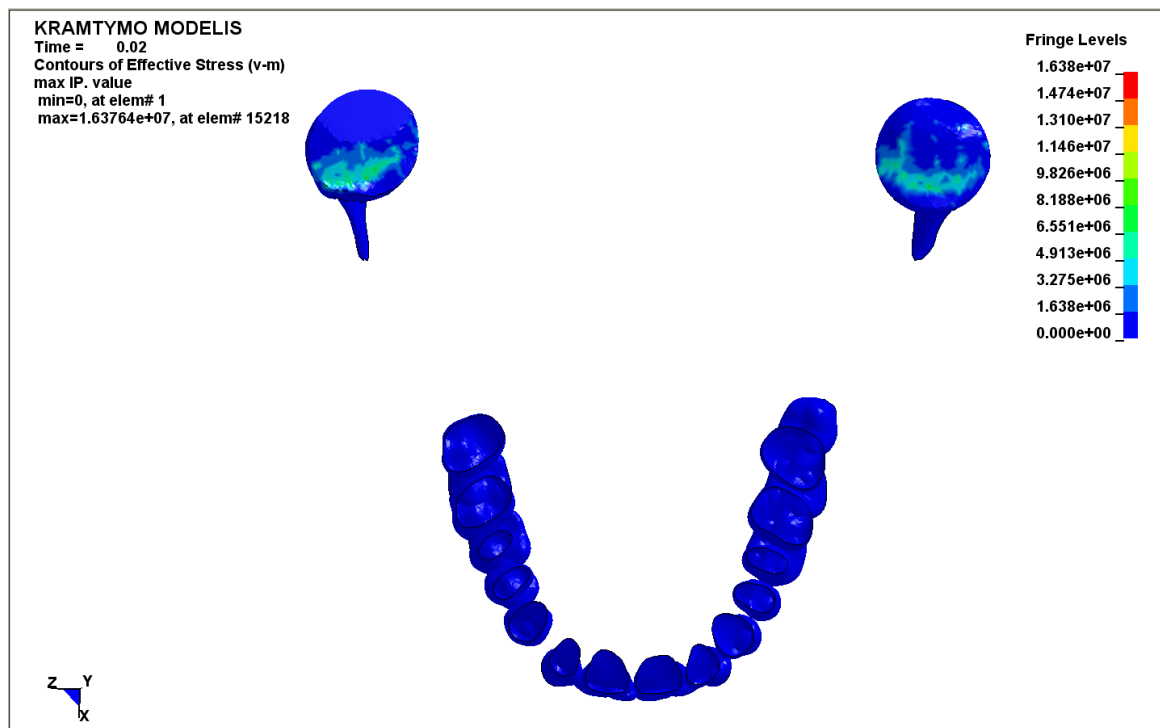
Pagrindinis vykdomasis failas. Jame geometrijos dalims priskiriamos medžiagų savybės, nusakoma kokios dalys kontaktuos ir t.t. Aukščiau aprašyti failai paleidžiami kartu su šiuo failu.

4. REZULTATAI

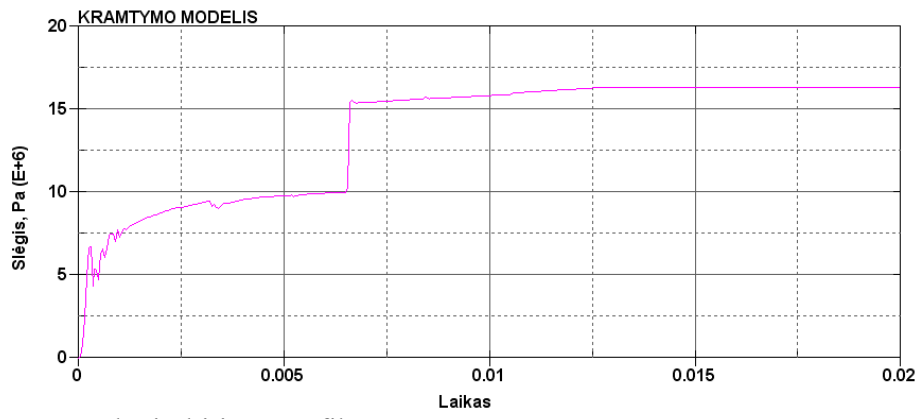
Paruošus keyword failus, beliko skaičiavimai LS-DYNA aplinkoje. Baigtinių elementų analizė su didesniu kiekiu elementų, reikalauja nemažų skaičiavimo pajėgumu. Skaičiavimai pastebimai paspartinami, jei procesorius (CPU) turi kuo daugiau branduolių. Baigtinių elementų analizė buvo atliekama su Intel Core 2 Duo 2.00 GHz procesoriumi, todėl rezultatai išaiškėjo tik kelių valandų begyje. Atlikus skaičiavimus paaiškėjo jog didžiausias slėgis į kremzlinius diskus yra 16,3 MPa. Žemiau esančiuose paveikslėliuose galima pastebėti, kuriose srityse pasiskirstęs slėgis. Slėgio reikšmės nustatomos pagal dešinėje pusėje esančia spalvine legenda.



21 pav. Kremzlinių diskų vaizdas iš apačios



22 pav. Kremzlinių diskų vaizdas iš viršaus



23 pav. Slėgio kitimo grafikas

5. IŠVADOS

1. Buvo sukurtas kompiuterinis 3D baigtinių elementų modelis susidedantis iš apatinio ir viršutinio žandikaulio dantų arkų, žandikaulinių duobučių ir kremzlinių sąnarių diskų.
2. Šis baigtinių elementų modelis parodė, kad kompiuterinis modeliavimas gali būti sėkmingai pritaikomas biomechaniniuose tyrimuose, tokiuose kaip slėgio pasiskirstymas kremzliniuose diskuose.
3. Modelio praktinė nauda yra ta, jog padeda įvertinti atsirandančias slėgio jėgas kremzliniuose diskuose, priklausomai nuo to koks pas žmogų yra sukandimas (skirtinga dantų forma)
4. Kad modelio geometrija būtų panašesnė į realią kramtymo sistemą, esamas „šurkštus“ paviršius buvo suglotnintas.
5. Sukurtas modelis ateityje gali būti panaudotas nagrinėjant kitas kramtymo sistemos dalys, kaip kad kiekvieno dančio atskirai nagrinėjimas.
6. Buvo atlikta keletas eksperimentų su kremzlinio sąnarinio disko medžiagos savybių parinkimu, kol buvo surastas tinkamas variantas.

6. LITERATŪROS SĄRAŠAS

G. Pileičikienė, A. Šurna, R. Barauskas, R. Šurna, A. Basevičius, Finite element analysis of stresses in the maxillary and mandibular dental arches and TMJ articular discs during clenching into maximum intercuspatation, anterior and unilateral posterior occlusion. 2007

http://www.techmat.vgtu.lt/konspektai/Skaicmet_netiesmech.html

<http://www.dynasupport.com/tutorial/contact-modeling-in-ls-dyna>

STL Files and Triangle Meshes. Prieiga per internetą: http://wiki.netfabb.com/STL_Files_and_Triangle_Meshes

Walton, D. J. and D. S. Meek, "A Triangular G1 Patch from Boundary Curves," Computer-Aided Design, Vol. 28 No. 2 pp. 113-123 (1996)

<http://www.dynasupport.com/manuals/lis-dyna-manuals/lis-dyna-971-manual-vol-i-keyword>

<http://www.dynasupport.com/manuals/additional/lis-dyna-theory-manual-2005-beta>

Andriani Daskalaki, Dental Computing and Applications: Advanced Techniques for Clinical Dentistry 2009, 170-190p

Wai Hon Wah, Introduction to STL format. 1999

Jianping Geng, Weiqi Yan, Wei Xu, Application of the Finite Element Method in Implant Dentistry

http://oras.if.ktu.lt/moduliai/P170m102/Literatura/KD_2.pdf