VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS INSTITUTAS
INFORMATIKOS KATEDRA

Kursinis darbas

Ciklinių koordinačių sistemų singuliarumų vizualizacija su OpenGL

(Visualization of singularities within cyclidic coordinate systems with OpenGL)

Atliko: 3 kurso 2 grupės studentas
Simas Mikelionis (parašas)

Darbo vadovas:
doc. dr. Rimvydas Krasauskas (parašas)

Vilnius
2022

Turinys

Įvadas

Šiais laikais, vizualizacijos mums padeda geriau apdoroti, įsisavinti ir suprasti informaciją. Tai yra ypač naudinga, norint pavaizduoti didelius kiekius duomenų vienu metu. Vienas iš pavydžių būtų grafikai, juose žymiai paprasčiau sekti duomenų pokytį nei žiūrint į skaičius lentelėse.

Dupino ciklinis paviršius yra specialus paviršius sukurtas Čarlzo Dupino 1803 metais. 1984 metais atgijo susidomėjimas Dupino cikliniais paviršiais ir jie buvo pradėti naudoti geometriniam modeliavimui. [Pra90]

Dupino paviršiai taip pat patogūs, norint atvaizduoti ortogonalią ciklinę koordinačių sistemą. Ši koordinačių sistema susidaro iš 3 ciklinių paviršių, kurie susikerta palei bendras kreives. Ciklinės koordinačių sistemos yra svarbus integruojamos sistemos pavyzdys. [Zak98].

Taigi šiame darbe aprašysiu Dupino skiautės ir ortogonalios ciklinės koordinačių sistemos bei jos singuliarių taškų kreivės vizualizaciją. Šią vizualizaciją implementuosiu ir pademonstriuosiu. **Darbo tikslas** – pavaizduoti interaktyvią ortogonalią ciklinę koordinačių sistemą ir jos singuliarių taškų kreives naudojant OpenGL vektorinės grafikos API.

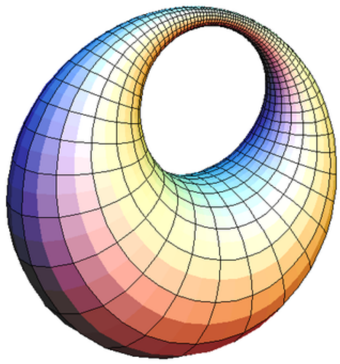
Darbo uždaviniai:

1. Išanalizuoti Dupino skiautės paviršiaus konstravimą.
2. Aprašyti Dupino skiautės paviršiaus vizualizavimą su OpenGL.
3. Aprašyti Dupino ciklinio kubo ir jo singuliarumų kreivių vizualizavimą su OpenGL.

1. Tyrinėjamos ciklinės koordinačių sistemos apibrėžimas

1.1. Dupino cikliniai paviršiai

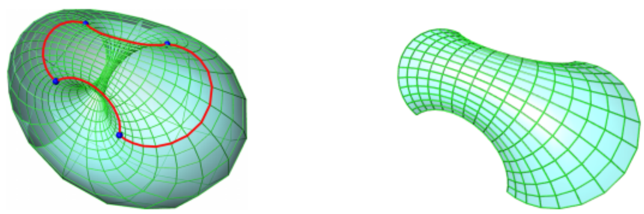
Dupino ciklinis paviršius pasižymi savo išlinkusiomis linijomis, kurios yra apskritimo dalis arba tiesė. [BH14] Šiame darbe pilni Dupino cikliniai paviršiai nebus vaizduojami, dėl to jų parametrizacija nebus apibrėžiama.



1 pav. Dupino ciklinis paviršius

1.2. Dupino skiautė

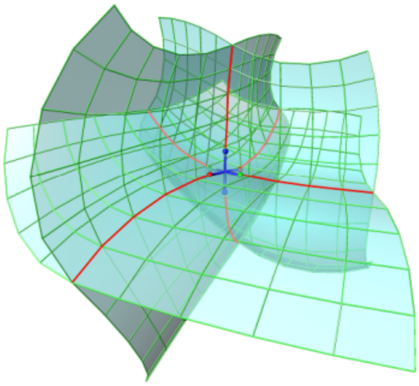
Dupino skiautė yra paviršius iškirptas iš Dupino ciklinio paviršiaus pagal išlinkimo linijas. Skiautė yra sudaroma iš 4 viršūnių p_1, p_2, p_3 ir p_4. Iš šių viršūnių yra sudaromos 4 kreivės $\widehat{p_1p_2}$, $\widehat{p_2p_3}$, $\widehat{p_3p_4}$, $\widehat{p_4p_1}$, kurios yra tiesės, arba apskritimo lankas. Taip pat, kiekvienas skiautės kampas turi liestinius vektorius $\vec{v_1}$ ir $\vec{v_2}$. Viršūnės p_i vektorius $\vec{v_1}$ liesis su kreive $\widehat{p_ip_j}$, o vektorius $\vec{v_2}$ liesis su kreive $\widehat{p_ip_k}$. Vektoriai $\vec{v_1}$ ir $\vec{v_2}$ yra statūs, jų vektorinė sandauga yra skiautės normalė n. [BH14]



2 pav. Dupino skiautė. [BH14]

1.3. Ciklinė koordinačių sistema

Šiame darbe bus vizualizuojama ortogonali ciklinė koordinačių sistemą Dupino skiautės paviršių pagalba. Koordinačių sistemos ortogonalūs paviršiai susidurs palei bendras kreives. Šios koordinačių sistemos vizualizavimas bus atliktas naudojant Dupino ciklinį kubą. [BH14]



3 pav. Ciklinė koordinačių sistema. [BH14]

1.4. Singuliarūs taškai

Koordinačių singuliarumas (angl. **Coordinate singularity**) - tai taškas koordinačių sistemoje, kuriame pasikeičia sistemos savybės arba atsiranda neapibrėžtumas (pvz. Polinėje koordinačių sistemoje taškas (0, α) ir (0, β) visada bus toje pačioje vietoje).

2. Vizualizacijos aprašymas

2.1. Naudota programinė įranga

Vizualizacijoms atlikti naudojamas programinis įrankis su C++ programavimo kalba ir OpenGL vektorinės grafikos API. Programoje taip pat naudojamos GLFW, GLM ir ImGui bibliotekos.

- **GLFW** - atviro kodo biblioteka, naudojama OpenGL, OpenGL ES ir Vulkan grafikos API naudojimui. GLFW pagrindinė paskirtis programoje yra atidaryti vaizdavimo langą bei jį kontroliuoti.

- **GLM** (OpenGL Mathematics) - yra matematikos biblioteka skirta OpenGL API.

- **ImGui** (Immediate Mode Grapchic User Interface) - lengva grafinės vartotojo sąsajos biblioteka. Ši biblioteka naudota dėl patogios įvesties bei išvesties programos veikimo metu, tai suteikia vizualizacijoms interaktyvumo.

Programoje naudojama perspektyvi kamera, kuri kontroliuojama naudojant vartotojo sąsają. Kamera galime pilnai apsukti aplink koordinačių centrą, taip pat ją galime pakelti ir nuleisti bei priartinti ir nutolinti nuo koordinačių centro.

Šioje programoje taip pat naudojamas kodas primityvioms geometrijoms generuoti (pvz. plokštumos geometrija, sferos geometrija ir vamzdžio geometrija). Geometrijos laiko viršūnių pozicijos, paviršiaus normalių ir paviršiaus UV koordinačių duomenis.

Šėšeliuoklė (angl. **Shader**) - tai kompiuterio programa, skirta skaičiavimams, reikalingiems 3D scenos vaizdavimui. Šiame darbe naudojamos viršūnių šėšeliuokles (angl. vertex shaders), erdvinių figūrų geometrijai apdoroti, ir pikselių šėšeliuokles (angl. fragment/pixel shaders), pikselių spalvinimui.

Tolygūs kintamieji (angl. **Uniform variables**) - tai kintamieji, paduodami į šėšeliuoklės programą. Jie veikia kaip vartotojo paduodami parametrai šėšeliuoklės programai. Šie kintamieji išlieka tokie patys vienam programos iškvietimui, dėl to jie vadinama tolygiais kintamaisiais.

Šėšeliuoklių programavimui buvo naudota GLSL (OpenGL Shading Language) programavimo kalba.

2.2. Kvaternijonai ir jų aritmetiniai veiksmai

Dupino skiaučių vizualizacijai naudojami kvaternijonų aritmetiniai veiksmai.
Įprastai, kvaternijonai reprezentuojami šia formule:

$$q = x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k} + w$$

Kvaternijonų produkto pagrindinės taisyklės:

$$i^2 = j^2 = k^2 = -1$$

$$ij = k$$

$$jk = i$$

$$ki = j$$

Kvaternijonai yra sudaryti iš dviejų dalių: realios - $Re(q) = w = r$ ir kompleksinės - $Co(q) = x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k} = \vec{v}$.
Kvaternijonų daugyba:

$$q_1q_2 = r_1r_2 - \vec{v}_1 \cdot \vec{v}_2 + r_1\vec{v}_2 + r_1\vec{v}_2 + \vec{v}_1 \times \vec{v}_2$$

Papildomi kvaternijonų žymėjimai:

- Kvaternijono konjugatas - $\overline{q} = r - \vec{v}$
- Kvaternijono ilgis - $|q| = \sqrt{r^2 + \vec{v} \cdot \vec{v}}$
- Atvirštinis kvaternijonas - $q^{-1} = \overline{q}/|q|^2$

Kvaternijonai šėšeliuoklėse bus laikomi keturmačiuose vektoriuose.

```
vec4 q = vec4(x, y, z, w);
```

2.3. Kvaternijoninė-Bézier kreivė

Kvaternijoninės-Bézier (KB) kreivės leidžia vaizduoti racionalias erdvines kreives. Kaip ir Bézier kreivėse, bus naudojamos linijines interpoliacijos, kurios leis gauti kvaternijoną tarp dviejų kitų kvaternijonų priklausomai nuo laiko t, kur $0 \leq t \leq 1$. Šiame darbe naudojamos tik pirmo laipsnio KB kreivės. Pirmo laipsnio KB kreivės visuomet sudarys apskritimo lanką. Kreivė sudaroma iš lanko galų p_1 ir p_2, ir taško, per kurį eina lankas q. [ZK15] Kreivės formulė atrodo taip:

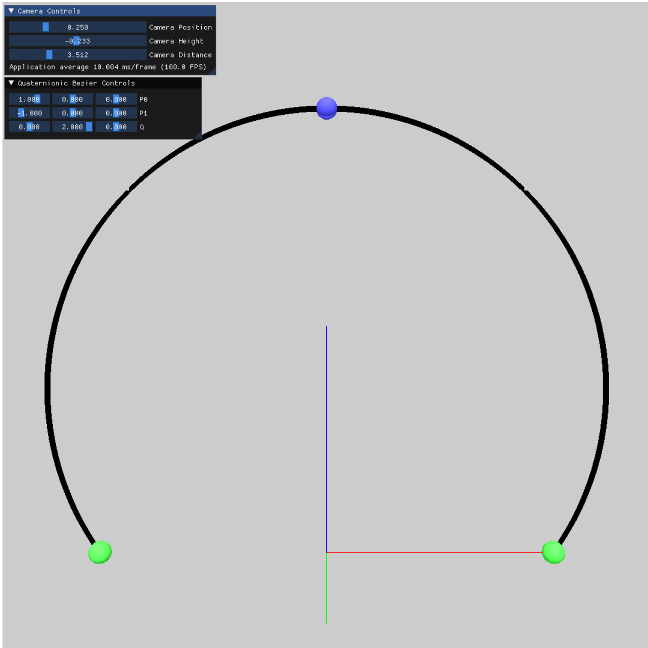
$$C(t) = (p_0w_0(1-t) + p_1w_1t)/(w_0(1-t) + w_1t)^{-1}$$

$$w_0 = (q - p_0)^{-1}$$

$$w_1 = (p_1 - q)^{-1}$$

Linijines interpoliacijas nuo šiol bus rašomos kaip OpenGL funkcija $mix(p_1, \quad p_2, \quad t)$. Kvaternijoninė-Bézier kreivė su šia funkcija atrodytų taip:

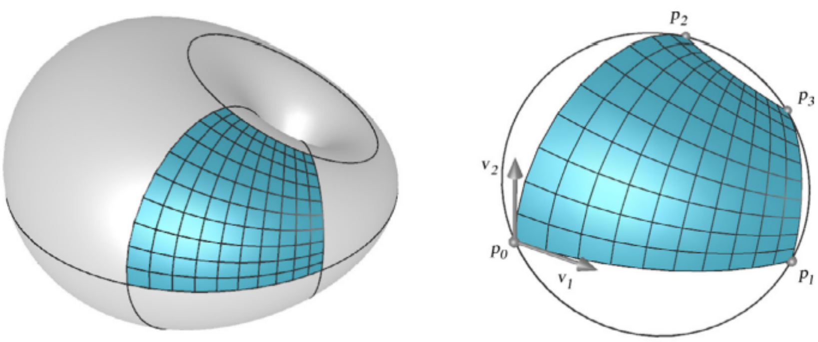
$$C(t) = mix(p_0w_0, p_1w_1, t)/(mix(w_0, w_1, t))^{-1}$$



4 pav. Kvaternijoninės-Bézier kreivės vizualizacija. Žalia spalva pažymėti taškai p_1 ir p_2, mėlyna spalva pažymėtas taškas q

2.4. Dupino skiautės vizualizacija

Dupino skiautė yra vizualizuojama deformuojant plokštumą pagal jos UV koordinates. Dupino skiautės šėšeliuoklė yra sudaryta iš viršūnių ir pikselių šėšeliuoklių.



5 pav. Dupino skiautės vizualizacija. [ZK15]

Į viršūnių šėšeliuoklę yra paduodama plokštumos geometrija, kuri laiko plokštumos viršūnių pozicijas, normales ir UV koordinates. Kaip tolygius kintamuosius, paduodame skiautės viršūnes p_0, p_1, p_2 ir p_3 bei viršūnės p_0 liestinius vektorius $\vec{v_1}$ ir $\vec{v_2}$, kur $\vec{v_1}$ bus $\widehat{p_0p_1}$ liestinė, o $\vec{v_2}$ bus $\widehat{p_0p_2}$ liestinė. Viršūnės ir vektoriai bus laikomi kaip kvaternijonai. Taip pat paduodama PV (angl. Projection View) matrica, kuri reikalinga 3D erdvės transformacijai į 2D ekrano erdvę.

Viršūnių šėšeliuoklėje pirmiausia apskaičiuojami taškų skirtumai naudojant formulę: [ZK15]

$$d_{ij} = p_i - p_j,\ kur\ i \neq j$$

Su šiais skirtumais galima apskaičiuoti kiekvieno skiautės taško svorius. [ZK15]

$$w_0 = 1, \quad w_1 = d_{10}v_2, \quad w_2 = d_{20}v_2 w_3 = -\frac{|d_{10}|}{|d_{32}|} * \frac{|d_{21}|}{|d_{30}|}(d_{32}d_{20})(v_1 v_2) \tag{1}$$

Toliau bus skaičiuojamos naujos viršūnės koordinatės pagal jos UV koordinates, naudojant Kvaternijoninę-Bézier kreivę. [ZK15]

$$skaitiklis = mix(mix(p_0w_0, p_1w_1, 1 - UV_x), mix(p_2w_2, p_3w_3, 1 - UV_x), UV_y)$$

$$vardiklis = mix(mix(w_0, w_1, 1 - UV_x), mix(w_2, w_3, 1 - UV_x), UV_y)$$

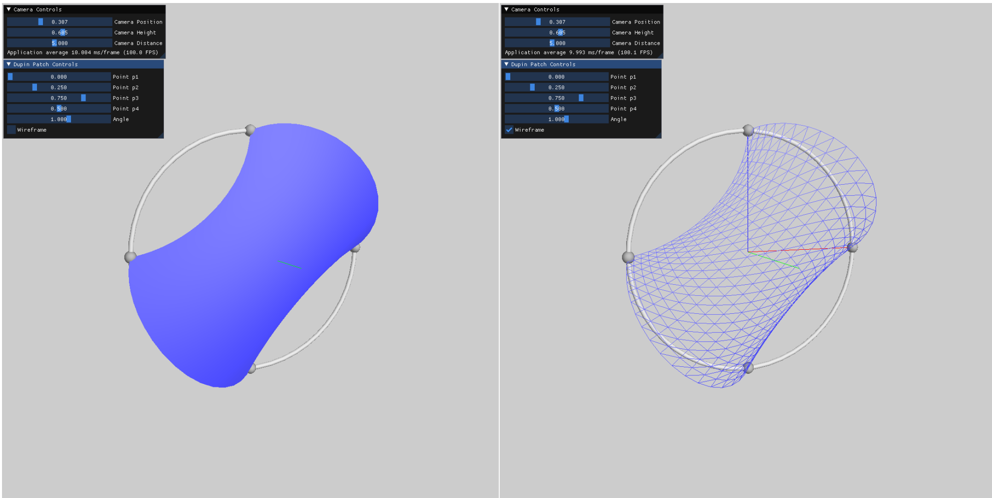
$$np = skaitiklis\ vardiklis^{-1}$$

Taškas np yra nauja plokštumos geometrijos viršūnės koordinatė. Liko apskaičiuoti tik normalę, tai galima padaryti atlikus kvaternijono posukį.

$$nor = vardiklis\ (\vec{v_1} \times \vec{v_2})\ vardiklis^{-1}$$

Viršūnių šėšeliuoklės išvestis yra naujai apskaičiuota viršūnės koordinatė, normalė ir UV koordinatės.

Pikselių šėšeliuoklės įvestis yra tokia pati, kaip viršūnių šėšeliuoklės išvestis. Taip pat paduodami tolygūs kintamieji - kameros pozicija, šviesos šaltinio pozicija, šviesos šaltinio spalva ir paviršiaus spalva. Šios šėšeliuoklės paskirtis yra pritaikyti Phong apšvietimą paviršiui. [Str11] Šis apšvietimas yra reikalingas dėl geresnio skiautės išlinkimo matomumo. Dar geresniam matomumui išgauti naudojamas tinklelio režimas, kurį galima kontroliuoti per vartotojo sąsają ir kuris atvaizduos skiautę kaip tinklelį.



6 pav. Dupino skiautės vizualizacija mano programoje. Dešinėje pusėje skiautė vaizduojama naudojant tinklelio režimą.

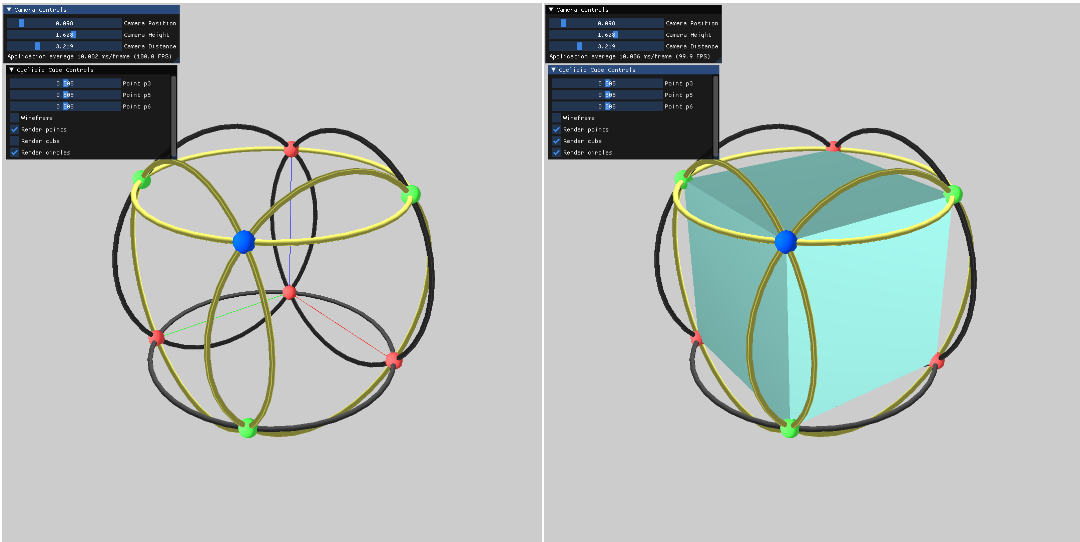
2.5. Dupino ciklinio kubo vizualizacija

Ciklinis kubas yra sudarytas iš šešių Dupino skiaučių: $S_A(p_0,\ p_2,\ p_4,\ p_6)$, $S_B(p_0,\ p_1,\ p_2,\ p_3)$, $S_C(p_0,\ p_1,\ p_4,\ p_5)$, $S_D(p_2,\ p_3,\ p_6,\ p_7)$, $S_E(p_4,\ p_5,\ p_6,\ p_7)$, $S_F(p_1,\ p_3,\ p_5,\ p_7)$. [BH14] Dupino kubo viršūnės p_0, p_1, p_2, p_4 turi pastovias koordinates:

$$p_0 = 0, \quad p_1 = k, \quad p_2 = i, \quad p_4 = j,$$

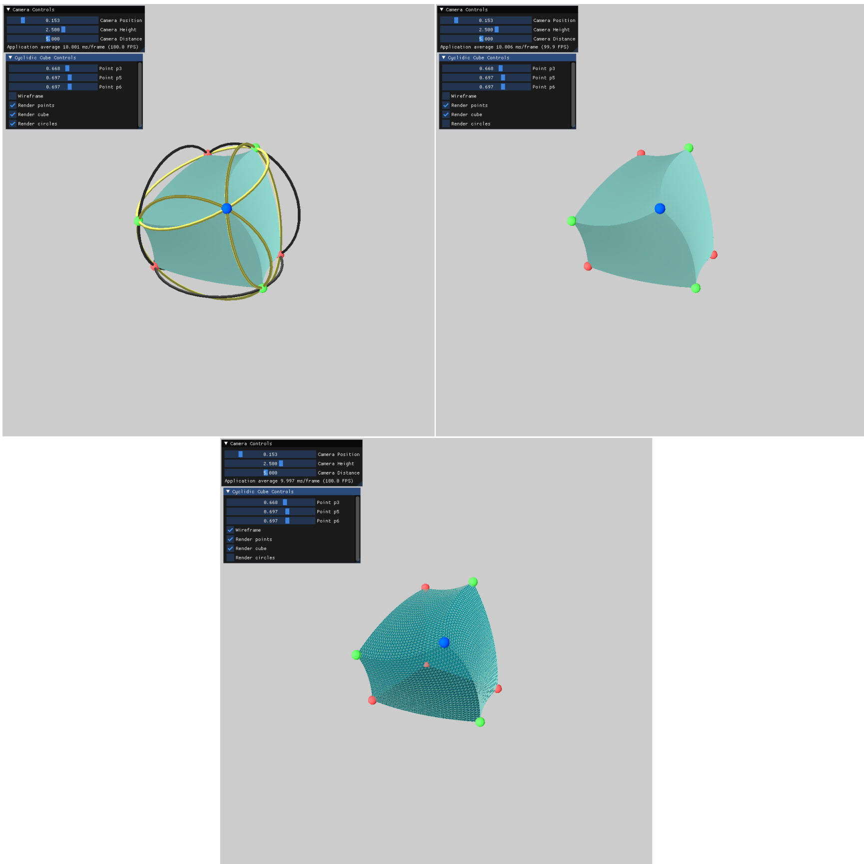
Viršūnės p_3, p_5, p_6 yra kontroliuojamos, jas galima judinti palei apskritimus, apskaičiuotus naudojant pastovias 3 viršūnes, priklausančias skiautėms S_A, S_B ir S_C. Viršūnes p_3, p_5, p_6 kontroliuojamos slankikliais a, b, c (0 $\leq a$,b,$c \leq$ 1), naudojant vartotojo sąsają. Slankiklio reikšmė a kontroliuoja viršūnę p_3, b - p_6 ir c - p_5.[BH14]

Kadangi plokštumos S_A, S_B ir S_C turi po tris pastovias viršūnes, jų apskritimai bus pastovūs, tačiau S_D, S_E ir S_F turi tik po vieną pastovią viršūnę ir po du kontroliuojamus taškus, dėl to jų apskritimai keisis, kai keičiasi p_3, p_5 ar p_6 pozicija. Viršūnė p_7 yra plokštumų S_D, S_E ir S_F susikirtimo taškas.[BH14]



7 pav. Dupino kubo viršūnės ir skiaučių apskritimai. Raudoni taškai vaizduoja pastovias viršūnes, žali taškai - kontroliuojamas, o mėlyni - apskaičiuojamą. Juodi apskritimai yra pastovūs, geltoni - kintantys.

Naudojant vartotojo sąsają taip pat galima išjungti ir įjungti viršūnių vizualizaciją, apskritimų vizualizaciją bei tinklelio režimą, kuris vaizduoja skiautes S_D, S_E, S_F kaip tinklelį.



8 pav. Dupino kubo vizualizacija.

2.6. Ciklinės koordinačių sistemos singuliarių taškų vizualizacija

Kadangi ciklinės koordinačių sistemos singuliarūs taškai yra koordinatės ašyse, juos galima atvaizduoti trimis plokštumomis. Šios plokštumos bus $P_1(x,\, y)$, $P_2(x,\, z)$, $P_3(y,\, z)$. Šias plokštumas reikia pasukti, kad jos atitiktų Dekarto koordinačių sistemos ašių plokštumas.

Su šiomis plokštumomis bus naudojama šėšeliuoklė, kuri nuspalvins singuliarių taškų kreivę ant plokštumos. Pikselių šėšeliuoklėje naudojamos per vartotojo sąsają kontroliuojamų viršūnių p_3, p_5, p_6 slankikliai a, b, c ir plokšumos S_E viršūnės p_4, p_5, p_6, p_7.
Pikselių šėšeliuoklėje skaičiuojami viršūnių svoriai, taip pat kaip ir Dupino skiaučių šėšeliuoklėse (1).
Apskaičiuoti singuliarių taškų kreives naudojamos numanomos formulės (angl. implicit equations). [ZK15] X bus kvaternijonas, laikantis UV koordinates. Singuliarių taškų kreivės formulė:

$$Eq = det((X - p_4)w_4, (X - p_5)w_5, (X - p_6)w_6, (X - p_7)w_7)$$

$$Eq_{xy} = 16\ Eq\ c^2\ (-1 + b)^2$$

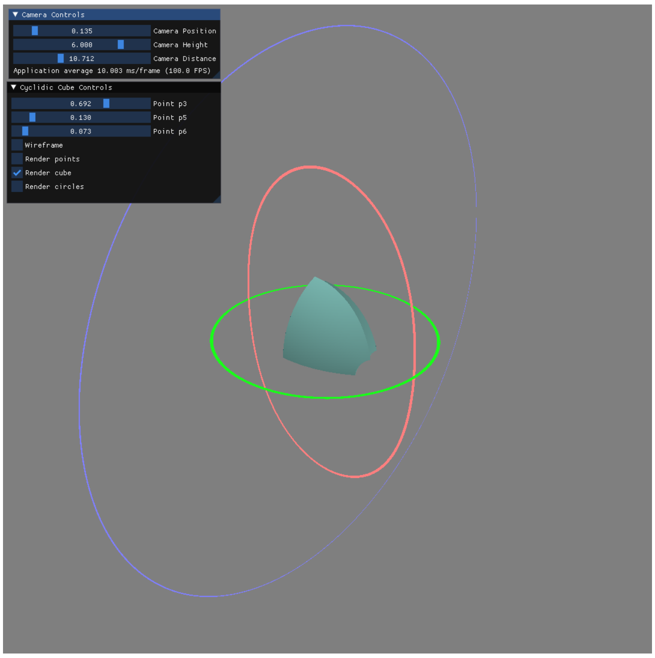
$$Eq_{xz} = 16\ Eq\ b^2\ (-1 + a)^2$$

$$Eq_{yz} = 16\ Eq\ a^2\ (-1 + c)^2$$

Plokštuma $P_1(x,\, y)$ naudos formulę Eq_{xy}, $P_2(x,\, z)$ naudos Eq_{xz}, o $P_1(y,\, z)$ - Eq_{yz}

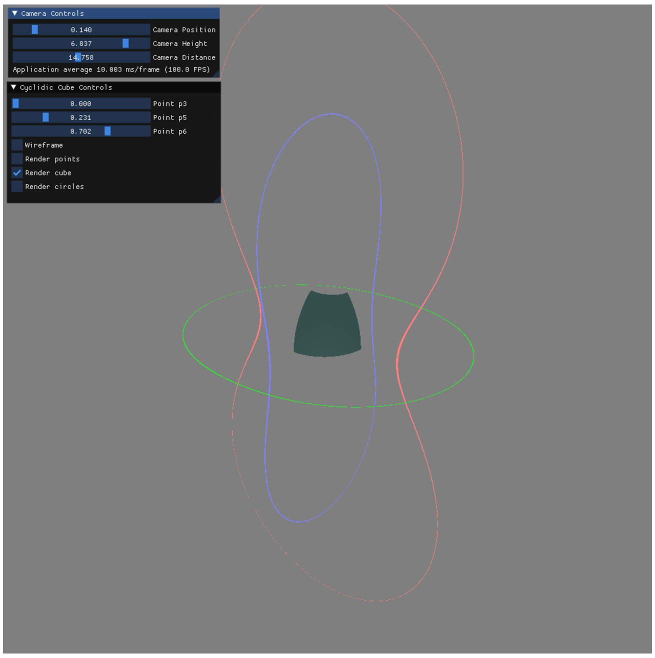
Singuliarių taškų kreivė spalvinama, kai formulių reikšmės yra 0. Programoje, dėl geresnio kreivių matomumo, jos piešiamos, kai jų reikšmės mažesnės už 0,035. Pikseliai, kurių koordinatėse formulės reikšmė yra daugiau 0,035, nepiešiami (naudojama $discard$ funkcija).

Singuliarių taškų kreivė $P1(x, y)$ bus spalvinama raudona spalva, kreivė $P2(x, z)$ - žalia, o $P3(y, z)$ - mėlyna.



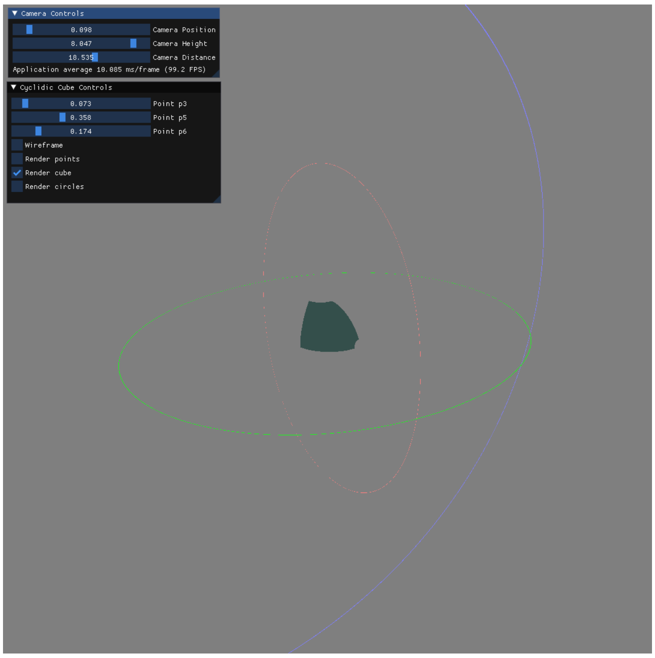
9 pav. Singuliarių taškų kreivės

Paveikslėlyje 9 matomos visos 3 singuliarių taškų kreivės. Viršūnių p_3, p_5 ir p_6 slankiklių reikšmės yra $a = 0.692$, $b = 0.130$, $c = 0.072$.



10 pav. Singuliarių taškų kreivės

Paveikslėlyje 10 matomos visos 3 singuliarių taškų kreivės. Viršūnių p_3, p_5 ir p_6 slankiklių reikšmės yra $a = 0.0$, $b = 0.231$, $c = 0.702$.



11 pav. Singuliarių taškų kreivės

Paveikslėlyje 11 matomos 3 singuliarių taškų kreivės. Viršūnių p_3, p_5 ir p_6 slankiklių reikšmės yra $a = 0.073$, $b = 0.358$, $c = 0.174$.

Išvados

Šiame darbe buvo aprašytas Dupino skiaučių konstravimas, bei vizualizacijos su OpenGL aprašymas. Aprašyta, kaip iš Dupino skiaučių galima sukonstruoti Dupino ciklinį kubą, kuris atvaizduoja ortogonalią ciklinę koordinačių sistemą.Taip pat aprašyta šios koordinačių sistemos singuliarių taškų kreivės vizualizaciją naudojant procedūrines tekstūras. Šiam darbui buvo parašyta programa naudojant OpenGL ir C++, kuri implementuoja šias vizualizacijas. Pasinaudojus šia programa, buvo pateiktos darbo uždavinio objektų vizualizacijos.

Pasiūlymai:

Šiuo metu, singuliarių taškų kreivės yra vaizduojamos plokštumose, dėl to, kai kamera yra ant kažkurios ašies, tos ašies kreivė yra nematoma.

Tinklelio rėžimas yra sunkiai matomas, kai plokštumos geometrija turi daug segmentų, nes tinklelio langeliai tampa labai maži ir sunkiai įžiūrimi. Būtų patogiau naudoti procedūrinę tinklelio tekstūrą, kuri nepriklausomai nuo plokštumos geometrijos segmentų skaičiaus, rodytų tą patį tinklelio dydį.

Literatūra

[BH14] Alexander I. Bobenko and Emanuel Huhnen-Venedey. Curvature line parametrized surfaces and orthogonal coordinate systems. discretization with dupin cyclides. 2014.

[Pra90] M.J. Pratt. Cyclides in computer aided geometric design. *Computer Aided Geometric Design*, 7(1):221–242, 1990.

[Str11] Paul S. Strauss. A realistic lighting model for computer animators. *Silicon Graphics*, 2011.

[Zak98] Vladimir E. Zakharov. Description of the n-orthogonal curvilinear coordinate systems and Hamiltonian integrable systems of hydrodynamic type, I: Integration of the Lamé equations. *Duke Mathematical Journal*, 94(1):103–139, 1998.

[ZK15] Severinas Zube and Rimvydas Krasauskas. Representation of dupin cyclides using quaternions. *Graphical Models*, 82:110–122, 2015.