



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
PANEVĖŽIO TECHNOLOGIJŲ IR VERSLO FAKULTETAS**

Mantas Kinderis

STIUARTO PLATFORMOS VALDYMO TYRIMAS

Baigiamasis magistro projektas

Vadovas

Prof. Vytenis Sinkevičius

PANEVĖŽYS, 2017

**KAUNO TECHNOLOGIJOS UNIVERSITETAS
PANEVĖŽIO TECHNOLOGIJŲ IR VERSLO FAKULTETAS**

STIUARTO PLATFORMOS VALDYMO TYRIMAS

Baigiamasis magistro projektas

Valdymo technologijos (kodas 621H66001)

Vadovas

Prof. Vytenis Sinkevičius

Recenzentas

Projektą atliko

Mantas Kinderis

PANEVĖŽYS, 2017



KAUNO TECHNOLOGIJOS UNIVERSITETAS
PANEVĖŽIO TECHNOLOGIJŲ IR VERSLO FAKULTETAS

(Fakultetas)

Mantas Kinderis

(Studento vardas, pavardė)

Valdymo technologijos, 621H66001

(Studijų programos pavadinimas, kodas)

„Stiuarto platformos valdymo tyrimas“

AKADEMINIO SAŽININGUMO DEKLARACIJA

20 m. d.

Panevėžys

Patvirtinu, kad mano, **Manto Kinderio**, baigiamasis projektas tema „**Stiuarto platformos valdymo tyrimas**“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

TVIRTINU:

BAIGIAMOJO PROJEKTO UŽDUOTIS

Išduota studentui: Mantui Kinderiui Grupė PME-5

1. Darbo tema:

Lietuvių kalba: Stiuarto platformos valdymo tyrimas

Anglų kalba: Research of Stewart Platform Control

Patvirtinta 2017 m. kovo mėn. 30 d. dekanu potvarkiu Nr. V25-13-8

2. Darbo tikslas:

Atlikti Stiuarto platformos valdymo galimybių tyrimą

3. Reikalavimai ir sąlygos:

Atlikti literatūros analizę. Išsiaiškinti tiesioginį ir atvirkštinį kinematikos uždavinį; Išsiaiškinti ir išmokyti kaip apskaičiuoti Stiuarto platformos atramos stovėjimo kryptis, panaudojant pritaikomas schemų lygtis. Sudaryti Stiuarto platformos ir jos hidraulinių servo vykdyklių modelį, naudojantis MATLAB programiniu paketu. Rasti optimalų vykdyklio kojų ilgį, reikalingą užduotims atlikti. Pateikti Stiuarto platformos scheminį vaizdavimą. Pateikti Stiuarto platformos judesių valdymo schemas; Pateikti Stiuarto platformos konstrukcijos laisvės laipsnių manipuliatorių judesio modeliavimo sistemas.

4. Projekto struktūra. Turinys konkretizuojamas kartu su vadovu, atsižvelgiant į BP pobūdį.

- 1. Stiuarto platformos struktūra, panaudojimo galimybės.*
- 2. Analizuojamo mechanizmo kinematika ir valdymo sistemos.*
- 3. Stiuarto platformos parametrų nustatymas, vykdyklių modeliavimas.*
- 4. Moduliatoriaus valdymo modeliavimas Simulink programos pakete.*
- 5. Moduliatoriaus parametrų tyrimas MatLab programos pakete.*
- 6. Moduliatoriaus valdymo schemų sudarymas ir jų pateikimas.*

5. Ši užduotis yra neatskiriama baigiamojo projekto dalis.

6. Projekto pateikimo gynimui kvalifikacinėje komisijoje terminas

2017-06-01

(data)

Užduotį gavau: Mantas Kinderis

(studento vardas, pavardė, parašas)

2017-02-01

(data)

Vadovas: Prof. Vytenis Sinkevičius

(pareigos, vardas, pavardė, parašas)

2017-02-01

(data)

TURINYS

IVADAS.....	8
1.1. Analizuojamo manipulatoriaus ištakos.....	9
1.2. Manipulatoriaus mechaninė sudėtis	11
1.3. Manipulatoriaus klasifikacija.....	11
1.4.1. Atvirkštinės kinematinės lygtys.....	12
1.4.2. Tiesioginis padėčių uždavinys.....	14
1.4.3. Jakobsiano matrica ir kinematinės greičių lygtys	15
1.4.4. Dinamikos lygtis.....	16
1.5. Valdymo sistemos	17
1.6. Manipulatoriaus pritaikymo galimybių analizė	18
1.6.1. Tikslus pozicionavimas.....	18
1.6.2. NASA LIDS (low impact docking system).....	19
1.6.3. Bangų stabilizavimo sistema laivo kranui.....	19
1.6.4. Lygiagrečios kinematinės struktūros kranai (RoboCrane).....	20
1.6.5. Medicinos sritis.....	20
1.7. Skyriaus išvados	22
2. METODOLOGINĖ DALIS	23
2.1. Stiuarto platformos parametrų nustatymas	23
2.2. Stiuarto platformos vykdyklių modeliavimas	27
2.3. Stiuarto platformos valdymo modeliavimas Simulink programoje.....	29
2.4. Stiuarto platformos modelis Simulink programoje	30
2.5. Stiuarto platformos dinaminį parametrų radimas	31
2.6. PID regulatoriaus parametrų nustatymas.....	32
2.7. Skyriaus išvados	32

3. TIRIAMOJI DALIS.....	34
3.1. Manipulatoriaus valdymas naudojant Simulink programos paketą.....	35
3.2. Matlab rogramos struktūros apibendrinimas.....	37
3.3. Programinis kodas.....	39
3.3.1. Pradinių reikšmių apskaičiavimas.....	39
3.3.2. Koordinačių ir kojų ilgių apskaičiavimas, bei jų kitimas laike.....	41
3.4. Kojų ilgių kitimo analizė.....	42
3.5. Kojų jėgų skaičiavimas.....	46
3.6 Skyriaus išvados.....	48
IŠVADOS	50
LITERATŪROS SĄRAŠAS.....	51
PRIEDAI.....	53

Kinderis, Mantas. Stiuarto platformos valdymo tyrimas. Magistro baigiamasis projektas / vadovas Prof. Vytenis Sinkevičius; Kauno technologijos universitetas, Panevėžio technologijų ir verslo fakultetas. Lietuvių kalba, 62 puslapiai, 40 iliustracijų, 28 formulės, 7 lentelės.

Mokslo kryptis ir sritis: Valdymo technologijos

Reikšminiai žodžiai: Stiuarto platforma, MatLab programa, lygiagretus manipulatorius, hidraulinė servo pavara, modelis, valdymas

Panevėžys, 2017. 59 p.

SANTRAUKA

Darbe nagrinėjama uždara mechaninė konstrukcija, kuri yra lygiagreti kinematinei grandinei ir turintis 6 laisvės laipsnius – Stiuarto platforma, jos moduliavimo ir valdymo galimybės. Analizuojama literatūra, kurioje esama informacija susijusi su turimo mechanizmo kinematika, klasifikacija pagal jo geometrines ir mechanines savybes, modeliavimo galimybes, dinamika pritaikymo galimybes įvairiose srityse.

Metodologinėje dalyje pateikiama modelio klasifikacija, panaudojimo galimybės, pasirinktos geometrinės struktūros išraiška atvirkštinės kinematikos lygtimis, valdymo sistemos, pagrįstos manipulatoriaus vykdyklių užduočių sudarymu, naudojant atvirkštinį padėčių uždavinį. Apžvelgiami Stiuarto platformos ir jo vykdyklių – hidraulinių servo pavarų sudarymo principai ir etapai panaudojant atvirkštines kinematinės lygtis bei esamos išanalizuotos manipulatoriaus matematinio modelio lygtys panaudojamos MATLAB programoje, tam kad būtų sudarytas manipulatoriaus modelis.

Tiriamąją dalį sudaro manipulatoriaus vykdyklio ilgio paieškos esant tam tikriems judesiams erdvėje. Rasti optimalūs Stiuarto platformos kojų ilgiai, kurie vėliau buvo panaudoti tolimesniam mechanizmo valdymo sistemos tyrimui. Gauti duomenys buvo panaudoti nustatyti vykdyklio pereinamojo proceso parametrus. Parametrus gauti buvo panaudoti konkrečios Stiuarto platformos turimi duomenys (kraštinių ilgiai, aukštis, svoris kurį platforma gali išlaikyti, pradiniai kojų ilgiai). Gauti eksperimentiniai duomenys pateikti lentelių ir grafikų pavidalu.

Kinderis, Mantas. Research of Stewart Platform Control / supervisor assoc. prof. Vytenis Sinkevičius. The Faculty of Panevėžys technology and business, Kaunas University of Technology. Lithuanian language, 62 pages, 40 images, 28 formulas, 7 tables.

Research area and field: Technologies of Control

Key words: Stewart platform, MatLab program, parallel manipulator, model, control
Panevėžys, 2017. 59 p.

SUMMARY

This thesis analyses closed mechanical structure, which is parallel to the kinematic chain and has 6 degrees of freedom, Stewart platform, possibilities of its modulation and control. Literature relating to the existing kinematics of the mechanism, the classification according to the geometrical and mechanical characteristics, modelling possibilities, the dynamics of the application opportunities in various fields is analysed.

Methodological part consists of model's classification, usability, expression of the selected geometric structure in inverse kinematics equations, control system based on actuators of manipulator tasks forming while using a reverse-way task. Furthermore, the principles of Stewart platform and forming of its' actuators - hydraulic servo actuators and steps of using the inverse kinematic equations are analysed in the thesis. The mathematical model equations of manipulator were used in MATLAB program in order to establish a model of the manipulator.

The analysis part includes the search of the manipulator actuator's length in certain movements. What is more, optimal length of Stewart platform legs was found. The lengths were used for further investigation of the control system of the mechanism. Data was used to determine the parameters of the actuator transition process. Data of specific Stewart platform (length of the sides, weight which can be kept by the platform and initial length of the legs) were used to find the parameters. The experimental data is presented in tables and figures.

IVADAS

Stiuarto platforma, kitaip dar vadinama Stiuarto-Gaufo platforma – tai specifinės uždarnos mechaninės konstrukcijos, kurios lygiagrečios kinematinei grandinei. [1] Stiuarto platformą 1954 m. sukūrė ir užpatentavo V. E. (Eric) Gough, o pirmosiose publikacijose 1965 m. paskelbė D. Stewart Jungtinės karalystės mechaninės inžinerijos institute (Institution of Mechanical Engineers) kaip šešis laisvės laipsnius turintį manipuliatorių, skirtą lėktuvų pilotams mokyti [1–2]. Šiais laikais pati platforma klasifikuojama pagal laisvės laipsnių skaičių ir sąnarų pavadinimų trumpinius, pagal manipulatoriaus kojų jungimosi šarnyrų su pagrindu ir judančia platforma skaičių. Priešingai nei nuoseklios kinematinės grandinės, Stiuarto platformoje apkrova darbinuose taškuose pasiskirsto tolygiau, o konstrukcijos standumas žymiai didesnis [2]. Būtent dėl tos teigiamos lygiagrečiųjų robotų savybės tokios konstrukcijos leidžia santykinai nedideliems lygiagretiesiems robotams manipuluoti didelėmis apkrovomis. Viena plačiausių pritaikymo sričių yra pozicionavimo sistemos [1–3].

Tiriamajame darbe pasirinkta 3-6 tipo Stiuarto platforma, kadangi tai bene labiausiai ir plačiausiai technikoje naudojamas manipulatoriaus tipas.

Tyrimo objektas: Stiuarto platforma.

Darbo tikslas: atlikti Stiuarto platformos valdymo galimybių tyrimą.

Tyrimo uždaviniai

- Atlikti literatūros, susijusios su Stiuarto platforma, jos modeliavimu, valdymu, ir pritaikymo galimybių analizę;
- Išsiaiškinti tiesioginį ir atvirkštinį kinematikos uždavinį; Išsiaiškinti ir išmokti kaip apskaičiuoti Stiuarto platformos atramos stovėjimo kryptis, panaudojant pritaikomas schemų lygtis. Sudaryti Stiuarto platformos ir jos hidraulinių servo vykdyklių modelį, naudojantis MATLAB programiniu paketu.
- Rasti optimalų vykdyklio kojų ilgį, reikalingą užduotiems erdvėje atlikti. Pateikti Stiuarto platformos scheminį vaizdavimą. Pateikti Stiuarto platformos judesio valdymo schemas.
- Pateikti Stiuarto platformos konstrukcijos laisvės laipsnių manipuliatorių judesio modeliavimo sistemas.

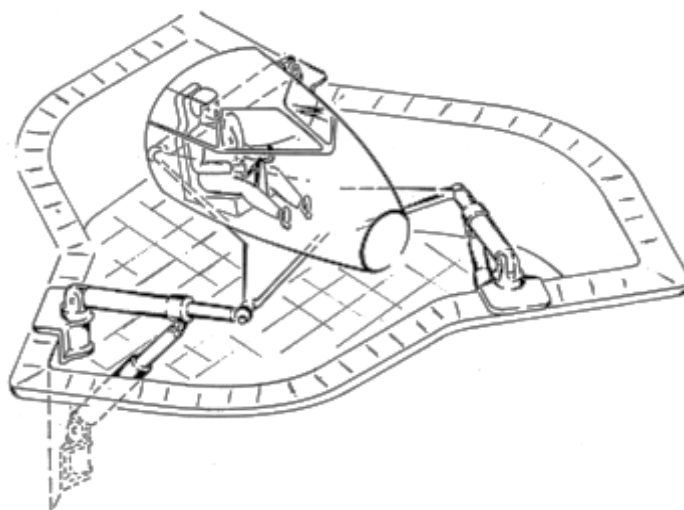
Tyrimo metodika: literatūros analizė, modeliavimas kompiuteriu.

1. ANALITINĖ DALIS

1.1. Analizuojamo manipulatoriaus ištakos

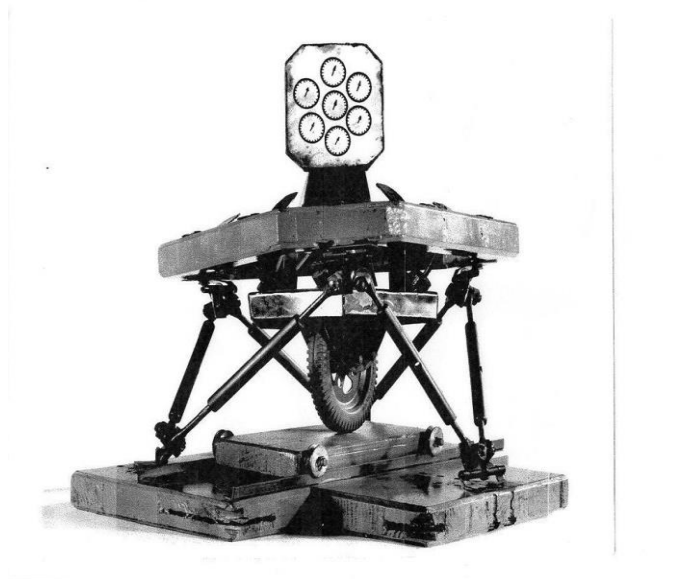
Stiuarto platforma (angl. Stewart-Gough) platforma) – tai uždaros mechaninės konstrukcijos, kurios yra lygiagrečios kinematinei grandinei. Ji naudojama kuriant, projektuojant ir paleidžiant judesiui robotams įvairiose srityse.

Ją 1954 m. sukūrė ir užpatentavo V. E. Gough. Jungtinės Karalystės mechaninės inžinerijos institutas (Institution of Mechanical Engineers) 1965 m. paskelbė, kad Stiuarto platforma šešis laisvės laipsnius turintis manipulatorius, kuris gali būti pritaikytas pilotams mokyti [4]. Sukurto manipulatoriaus konstrukciją sudarė judanti trikampė platforma arba kabina sujungta su nejudančiu stabiliu pagrindu per tris platformos kojas (gembes), kurių kiekviena sudaryta iš dviejų linijinių pavarų (1 pav). Abi linijinės pavaros su manipulatoriaus pagrindu sujungtos per dviejų ašių sąnarus (jos gali būti : rotacinės, prizmatinės, cilindrinės, universaliosios ir sferinės). Apatinė linijinė pavara su viršutiniąja sujungta per vienos ašies sąnarą, o judanti platforma su viršutiniąja koja sujungta per trijų ašių sąnarą. Būtent tokia esama konstrukcija leidžia turimą manipuliatorių valdyti polinėje koordinatų sistemoje, kuomet viršutinė platforma juda apatinės stabilios platformos atžvilgiu tam tikrose ribose.

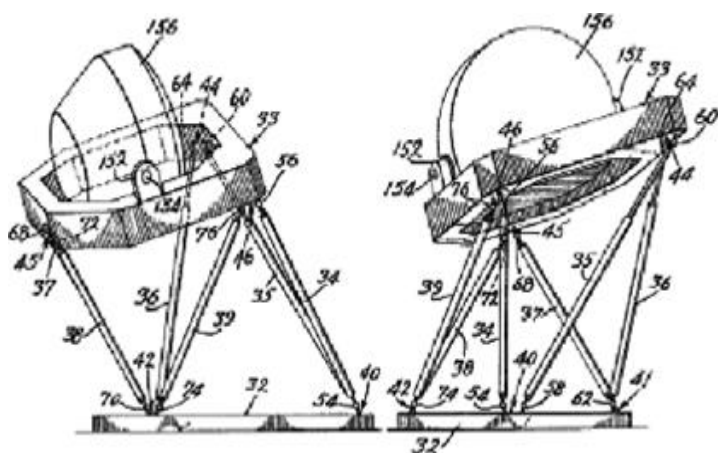


1 pav. Stiuarto platformos modelis [5]

Pirmieji mokslininkai, pritaikę Stiuarto platformos konstrukciją buvo E. Gough ir K. Cappel. Jie kaip ir pats Stiuartas manipuliatorių pritaikė pilotų mokymo institucijose.



2 pav. E.Gough manipulatorius



3 pav. K. Cappel manipulatorius

E. Gough sukurtas manipulatorius buvo skirtas padangų bandymams(2 pav.), o K. Cappel buvo pirmasis, kuris užpatentavo 6 laisvės laipsnių skrydžio simulatorių (3pav.) Laikui bėgant tokia manipulatoriaus koncepcija buvo pritaikyta plačiau ir įvairiose mokslo srityse.

1.2. Manipulatoriaus mechaninė sudėtis

Stiuarto platformai judesį suteikia šešios linijinės pavaros, kurios gali būti:

- pneumatinės,
- hidraulinės,
- elektromechaninės,
- magnetinės,
- pjezoelektrinės ir pan.

Kinematiškai šios pavaros vaizduojamos kaip cilindrinės arba prizmatinės sąnaros, turinčios linijinį poslinkį viena ašimi. Populiariausios yra elektromechaninės sraigtinės ir hidraulinės linijinės pavaros, tačiau neatsisakoma servo pavarų, kurios nuo seno įprastos ir gerai žinomos. Servo pavaros – tai sukamosios pavaros, kurios leidžia tiksliai kontroliuoti kampinę padėtį, sukimosi greitį ir pagreitį. Valdymą atliekanti dalis turi būti specialiai suprojektuota būtent to tipo servo varikliui. Šios pavaros išsiskiria tuo, jog naudoja grįžtamąjį ryšį.

Platformoje esamų kojų grandys apibūdinamos jas sudarančiomis sąnaromis ir žymimos sąnarų sutrumpinimais (pavyzdžiui UPS grandį sudaro universalioji, prizmatinė ir sferinė sąnaros).

Kojos grandies sudarymui naudojamos tokios sąnaros:

- rotacinės,
- prizmatinės,
- cilindrinės,
- universaliosios,
- sferinės.

Praktikoje, kuriant/projektuojant robotus dažniausiai sutinkamos universaliosios, sferinės ir prizmatinės arba cilindrinės sąnaros. [5-6]

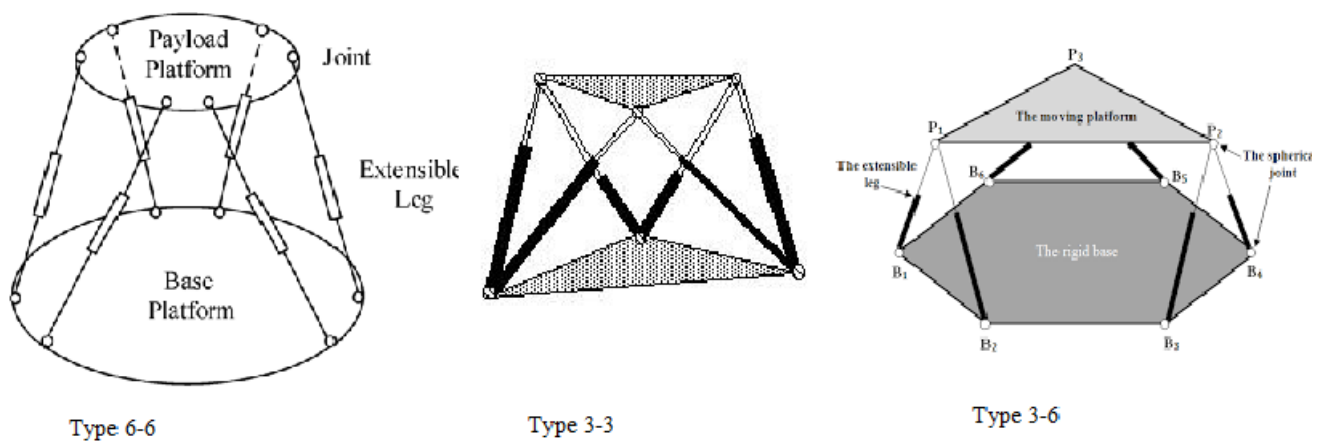
1.3. Manipulatoriaus klasifikacija

Šiais laikais platformos, remiantis mechanine architektūra ir galiojančiomis dviem *de facto* sistemomis klasifikuojama pagal:

- laisvės laipsnių skaičių ir sąnarų pavadinimų trumpinius;
- manipulatoriaus kojų jungimosi su pagrindu ir judančia platforma skaičių.

Taigi, tas pats manipulatorius gali būti apibūdintas dviem būdais, pvz., manipulatorius su šešiomis kojomis, turintis 6 laisvės laipsnius, kurio kojos yra prizmatinės sąnaros, su pagrindo platforma jos jungiamos universaliosiomis sąnaromis trijuose taškuose, o su judančia platforma sferinėmis sąnaromis taip pat trijuose taškuose.

Dažniausiai naudojamos geometrinės konfigūracijos yra 3-3, 3-6 ir 6-6, rečiau galima sutikti 6-3 konfigūraciją, galimos ir tarpinės konfigūracijos iš 3, 4, 5 ir 6 jungimosi taškų (4pav)



4 pav. Klasifikavimo pavyzdžiai

Nepaisant pasirinkto manipulatoriaus tipo, už kitas lygegretaus jungimo grandinės robotus Stiuarto platforma pranašesnė savo standumu, svorio ir išvystomos jėgos santykiu bei tikslumu. Tačiau ji turi nedidelę darbinę erdvę bei tam tikroje darbo erdvėje egzistuojančius neapibrėžtumus, kuriuose robotas tampa nevaldomas.

1.4.1. Atvirkštinės kinematinės lygtys

Kadangi tiriamoji Stiuarto platforma valdoma specifiniu manipulatoriumi, todėl tam jų juo būtų galima naudotis turi būti panaudojama **atvirkštinė kinematinė lygtis**. Būtent todėl:

manipulatoriaus pagrindo platformos koordinačių sistema sutapatinama su absoliučiąja koordinačių sistema;

aprašoma manipulatoriaus pagrindo ir judančios platformų kojų sujungimo taškų padėtis koordinačių pradžios taško atžvilgiu.

Pagrindo (nejudančios) ir judančios platformos atitinkamai atskirų taškų koordinatės surašomos į vektorines matricas \mathbf{B}_i ir \mathbf{P}_i ($i = 1 \dots 6$). Judriosios platformos judėjimui ir valdymui pačioje erdvėje reikalinga posūkio matrica \mathbf{R} . Ši matrica yra kompleksinė 3×3 kvadratinė, jos vertė

priklauso nuo Oilerio posūkių kampų α (alfa), β (beta), γ (gama) apie X, Y, Z ašis. Matrica \mathbf{R} nurodo judančios platformos koordinatinių sistemos posūkį absoliučios koordinatinių sistemos pradžios atžvilgiu (1)

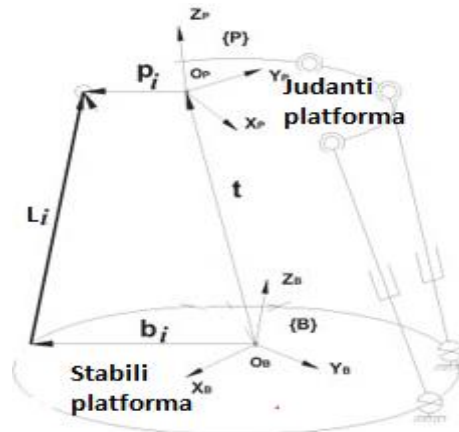
$$\mathbf{R} = \begin{bmatrix} c\alpha \cdot c\beta & s\alpha \cdot s\beta \cdot s\gamma - s\alpha \cdot c\gamma & c\alpha \cdot s\beta \cdot c\gamma + s\alpha \cdot s\gamma \\ s\alpha \cdot c\beta & s\alpha \cdot s\beta \cdot s\gamma + c\alpha \cdot c\gamma & s\alpha \cdot s\beta \cdot c\gamma - c\alpha \cdot s\gamma \\ -s\beta & c\beta \cdot s\gamma & c\beta \cdot c\gamma \end{bmatrix}, \quad (1)$$

Čia: $sx = \sin(x)$; $cx = \cos(x)$; x – vienas iš Oilerio posūkių kampų α, β, γ

Kita labai svarbi matrica – padėties matrica \mathbf{T} , ji nurodo judančios platformos koordinatinių sistemos pradžios taško O' padėtį absoliučioje koordinatinių sistemoje. Ši matrica susieja abiejų platformų (judriosios ir stabiliosios) vidurio taškus (2) [7].

$$\mathbf{T} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}, \quad (2)$$

čia x, y ir z – taško O' koordinatės Dekarto koordinatinių sistemoje.



5 pav. Vektoriai, naudojami atvirkštiniam padėčių uždaviniui spręsti

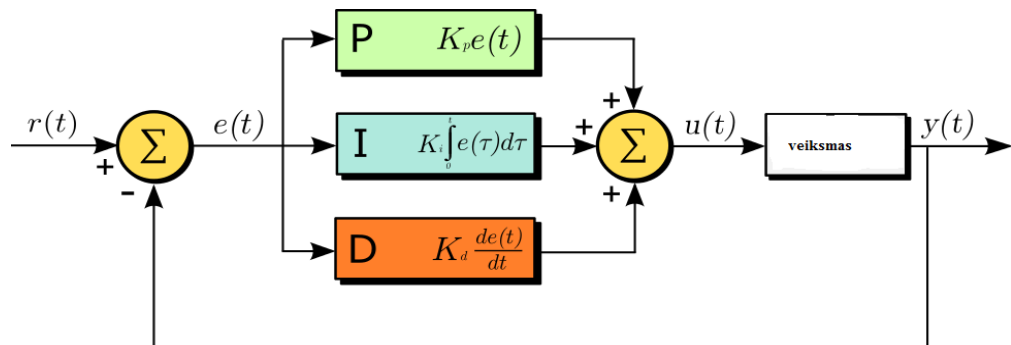
Posūkių ir padėties matricos panaudotos apskaičiuoti analizuojamo modulatoriaus kojų ilgiui:

$$L_i = \mathbf{R} \cdot \mathbf{p}_i + \mathbf{T} - \mathbf{b}_i \quad (i = 1, \dots, 6), \quad (3)$$

$$[L_i] = \begin{bmatrix} R \\ R \\ R \end{bmatrix} \cdot \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} + \begin{bmatrix} T \\ T \\ T \end{bmatrix} - \begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix}, \quad (4)$$

čia: L – kojos ilgis, išreikštas vektoriais; i – kojos numeris (1,...,6), R – judėjimo matrica p_i -viršutinės (judančios) platformos kojų koordinatės, T – padėties koordinatės, b_i – apatinės / stabiliosios platformos kojų taškų koordinatės.

Sudarius lygčių sistemas kojų ilgiams apskaičiuoti, ją galima naudoti turimos platformos trajektorijai numatyti. Iš šių atvirkštinių kinematikos lygčių gautos manipulatoriaus padėties perduodamos reguliatoriui. Dažniausiai tai PID reguliatoriai – plačiausiai paplitę, turintys grįžtamąjį ryšį ir naudojami pramoninėse kontrolės sistemose. Jį sudaro 3 grandys: proporcinė (P), integruojanti (I) ir diferencijuojanti (D).



6 pav. PID reguliatoriaus schema [8]

Tačiau gali būti naudojami ir PD reguliatoriai [8] bei netiesiniai PID reguliatoriai [8] ir observatoriai [9].

1.4.2. Tiesioginis padėčių uždavinys

Analizuojant Stiuarto platformą taikomas tiesioginis padėčių uždavinys – tai sudėtingos netiesinės lygčių sistemos, turinčios daug galimų sprendimų. Tiesioginis padėčių uždavinys lygiagrečiam manipulatoriui - tai 18 netiesinių lygčių sistema su 6 nežinomaisiais, gautas sprendinys – viršutinės judančios platformos pradžios taškas koordinačių sistemoje ir manipulatoriaus posūkis esmos absoliučios koordinačių sistemos atžvilgiu. Lygiagretieji manipulatoriai gali turėti nuo 12 iki 40 realių sprendinių [9-10], tačiau dalis jų gali būti atmetami dėl savo negalėjimo realizuoti. Dauguma darbų, kuriuose analizuojami tiesioginiai padėčių uždaviniai lygiagrečiuose manipulatoriuose, paremti geometrine manipulatoriaus kinematinės struktūros analize (pavyzdžiui Niutono-Rafsono metodas). Esamas tiesioginis kinematikos uždavinys leidžia apskaičiuoti apibendrintąsias judančios platformos koordinates. Jos reikalingos

valdymo sistemai, apskaičiuojant manipulatoriaus darbinės erdvės ribas bei judėjimo galimybes. Lygtis gali būti išvedama iš (3) lygties, kuomet ieškoma nežinomų poslinkio \mathbf{T} ir posūkio \mathbf{R} matricių. Dažniausiai tiesioginei kinematikos lygčiai spręsti taikomas Niutono-Rafsono metodas [5]. Tiesioginės kinematinės lygties sprendiniams rasti naudojama (5) formulė:

$$\mathbf{q}_{j+1} = \mathbf{q}_j + \mathbf{J}_{l_0}^{-1} \cdot (l_0 - l_j) \quad (5)$$

čia: \mathbf{q} – manipulatoriaus judančios platformos apibendrintosios koordinatės, \mathbf{J} – Jakobiano matrica, l_0 – pradinis vykdiklio ilgio spėjimas, l_j – vykdiklio ilgio spėjimas j -tosios iteracijos metu.

Tačiau Niutono -Rafsono lygtis turi trūkumą, kadangi šiuo atveju apskaičiuojamas tik dalinis sprendinys, esantis arčiausiai tikslo. C. Yang (2009) savo darbe [4] aprašė GNRMD (angl. Global

Newton – Raphson with Monotonic Descent Algorithms) – tai algoritmas skirtas apskaičiuoti sprendinius realiu laiku. Anot šio autoriaus būtent GNRMD algoritmai pašalina Niutono – Rafsono buvusius trūkumus, papildant monotoninio nuolydžio operatoriumi w , kuris algoritmui yra:

$$\mathbf{q}_{j+1} = \mathbf{q}_j + w_j \cdot \mathbf{J}_{l_0}^{-1} \int_{\Theta_j} (l_0 - l_j), j = 0,1,2,\dots \quad (6)$$

čia w_j - monotoninio nuolydžio operatorius kiekvienoje iteracijoje, $0 < w \leq 1$.

Sukurta aldymo sistema su papildytu GNRMD algoritmu pritaikyta pramoniniui kompiuteriui realiu laiku ir sukurta programa parašyta ir paleista MATLAB programos paketu. Šiuo atveju pats algoritmas apribotas tokiais sąlygomis:

$$\begin{cases} \max \{ \|h(\mathbf{q}_j)\| \} < \varepsilon \\ j \leq N \end{cases} \quad (7)$$

čia ε – ribinė tolerancija, N – didžiausias iteracijų skaičius.

1.4.3. Jakobsiano matrica ir kinematinės greičių lygtys

Analizuotoje literatūroje Jakobianas – tai dalinių išvestinių matrica, susiejanti grandžių greičius pasirinktoje koordinačių sistemose su judančios platformos greičiu absoliučioje koordinačių sistemoje. Atvirkštinė Jakobiano lygtis randama diferencijuojant jau anksčiau minėtą atvirkštinę kinematinę (3) lygtį laiko atžvilgiu. Tai atlikus surišamas vykdiklių greitis \mathbf{q}' su platformos greičiu \mathbf{X}' :

$$\frac{\delta \dot{\mathbf{q}}}{\delta t} = \mathbf{J} \cdot \frac{\delta \mathbf{X}}{\delta t} \rightarrow \dot{\mathbf{q}} = \mathbf{J} \cdot \dot{\mathbf{X}} \quad (8)$$

čia: $\mathbf{X} = [x, y, z, \psi, \theta, \varphi]^T$ – apibendrintas padėties vektorius absoliučioje koordinačių sistemoje, $\dot{\mathbf{X}} = [v, \omega]^T$ – apibendrintas greičių vektorius.

Išreiškus kojos vektorius jo ilgio skaliaro \mathbf{q}_i ir krypties vektoriaus \mathbf{e}_i sandauga, ir (3) formulės išvedama lygtis naudojama Jakobiano apskaičiavimui:

$$\dot{\mathbf{q}}_i = \mathbf{e}_i \cdot \mathbf{v} + (\mathbf{R} \cdot \mathbf{a}_i \times \mathbf{e}_i) \cdot \boldsymbol{\omega} \quad (9)$$

Toliau ši lygtis panaudojama sudarant 6x6 Jakobiano matricą:

$$\mathbf{J} = \begin{bmatrix} \mathbf{e}_1^T & (\mathbf{R} \cdot \mathbf{a}_1 \times \mathbf{e}_1)^T \\ \dots & \dots \\ \mathbf{e}_6^T & (\mathbf{R} \cdot \mathbf{a}_6 \times \mathbf{e}_6)^T \end{bmatrix} \quad (10)$$

Ši matrica padeda apskaičiuoti judančios platformos kitimo greitį vektorinėje sistemoje, žinant vykdyklių padėčių kitimo greičius. Taip pat ji naudojama rasti manipulatoriaus sritis, kuriose jis tampa nevaldomu.

Nevaldomumas randamas, skaičiuojant Jakobiano dedamųjų determinantus ir yra dvi galimybės kada analizuojamas mudulatorius gali būti nevaldomi:

- jei \mathbf{J}_q dedamosios determinantas yra lygus 0, manipulatorius praranda vieną laisvės laipsnį ir tampa nevaldomas.
- jei \mathbf{J}_x dedamosios determinantas yra lygus 0, manipulatorius įgauna vieną papildomą laisvės laipsnį ir taip tampa nevaldomas.

Kadangi Jakobianas yra kaip ir vienetinė matrica, todėl galimas tik vieno tipo nevaldomumo atsiradimas.

1.4.4. Dinamikos lygtis

Norint sudaryti manipulatoriaus matematinį modelį sudaromos dinamikos lygtys. Pats manipulatoriaus matematinis modelis panaudojamas valdymo sistemose, taip įvertinant veikiančias jėgas, judesius. Dinamikos lygtys ir modelis gali būti išvestos naudojantis Lagranžo, Niutono-Oilerio ir Keino metodais.

Lagranžo metodas paremtas analizuojamo roboto dekompozicija į judančią platformą ir roboto kojas(tokiu atveju apskaičiuojamos kinetinės ir potencinės energijos- linijiniams ir rotaciniams judesiams) [11].

Lagranžo formuluotė apibendrintoms koordinatėms :

$$\frac{d}{dt} \cdot \frac{dL}{d\dot{\mathbf{q}}} - \frac{\delta L}{\delta \mathbf{q}} = \frac{d}{dt} \left(\frac{\delta K(\mathbf{q}, \dot{\mathbf{q}})}{\delta \dot{\mathbf{q}}} \right) - \frac{\delta K(\mathbf{q}, \dot{\mathbf{q}})}{\delta \mathbf{q}} + \frac{\delta P(\mathbf{q})}{\delta \mathbf{q}} = \boldsymbol{\tau} \quad (11)$$

čia: $\boldsymbol{\tau}$ – apibendrintose koordinatėse veikiančių jėgų matrica, $\mathbf{K}(\mathbf{q}, \dot{\mathbf{q}})$ – kinetinė energija, $\mathbf{P}(\mathbf{q})$ – potencinė energija.

Pakeitus koordinates absoliučiosiomis Dekarto koordinatėmis, gaunama patobulinta Lagranžo formulė:

$$\mathbf{J}^T(\mathbf{X})\mathbf{F} = \mathbf{M}(\mathbf{X})\ddot{\mathbf{X}} + \mathbf{C}(\mathbf{X}, \dot{\mathbf{X}})\dot{\mathbf{X}} + \mathbf{G}(\mathbf{X}) \quad (12)$$

čia: \mathbf{J} – Jakobiano matrica, \mathbf{F} – vykdyklio generuojama jėga veikianti i-tojoje manipulatoriaus kojoje, kojos sąnarus ašies kryptimi nukreipta į judančią platformą, \mathbf{M} – platformos masių matrica, \mathbf{C} – Koriolio ir išcentrinių jėgų matrica, \mathbf{G} – gravitacinių jėgų matrica.

Ši ir kitos dinamikos lygtys gali būti naudojamos matematiniam platformos sudaryti, tam geriausiai tinka programinis paketas MATLAB .

1.5. Valdymo sistemos

Pagrindinis valdymo sistemos uždavinys užtikrinti analizuojamo manipulatoriaus iškelto uždavinio vykdymą didžiausiu kieku įmanoma tikslumu, atsižvelgiant į galimus nukrypimus ir trikdžių kompensavimą [12]. Stiuarto platforma, ne išimtis, ji valdoma absoliučiose koordinatinių sistemose, kurios sujungtos kinematinėmis lygtimis. Būtent todėl tam kad valdytume Stiuarto platformą, valdymo sistemoje būtina spręsti kinematinis uždavinius jau minėtoms koordinatinių sistemų reikšmėms apskaičiuoti.

Paprasčiausias būdas valdyti Stiuarto platformą Dekarto koordinatinių sistemoje yra atvirkštinio kinematikos uždavinio sprendimas. Šios judesio valdymo sistemos pagrįstos tiesiniu pavienių grandžių valdymu, naudojant proporcingus – P, integralinius – I ir diferencialinius - D reguliatorius, bei jų kompleksines atmainas.

Tokias valdymo sistemas galima skirti į šias grupes:

- greičio (padėties);
- jėgos valdymo.

Dažniausiai greičio valdymui naudojamas PI reguliatorius, kurio veikimas pagrįstas grįžtamoju ryšiu.

Greičio reguliavimas leidžia kontroliuoti apkrovos švytavimus ir padėties grįžtamasis ryšys kontroliuoja dinaminį atsaką (nusistovėjimo laiką). Tiriamos grandies judesio greičio reguliatorius geriausiai išnaudojamas pereinamojo proceso metu, kai grįžtamojo ryšio atsakas yra lėtas.

Kitas naudojamas valdymo būdas – tai tiesinis proporcinis-diferencialinis PD jėgos valdymas, jis tikslenis nei prieš tai minėtas valdymo būdas.

Netiesinės valdymo sistemos taikomos tada, kai tiesinių valdymo sistemų nepakanka užtikrinti dinaminiam manipulatoriaus tikslumui. Jos naudoja išvestus dinامينius mechanizmo matematinius modelius valdymo grandyje. Esamo manipulatoriaus valdiklis gali būti paremtas tiesiniu PID reguliatoriumi komponuojant su prediktyviu, adaptyviu ar kitu dar sudėtingesniu reguliatoriumi.

1.6. Manipulatoriaus pritaikymo galimybių analizė

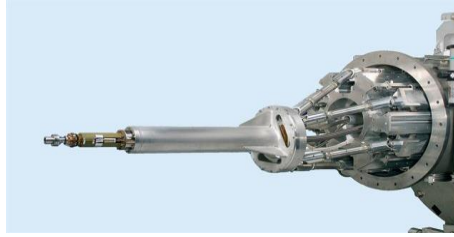
Pačioje pradžioje Stiuarto platforma buvo sukurta kaip šešis laisvės laipsnius turintis manipulatorius, kuris turėjo būti pritaikytas pilotų mokymui, kaip skrydžio simulatorius. Ji prieš nuosekliosios kinematikos grandines turi kelis pranašumus:

- lygiagrečiai kinematinė grandinė priešingai nei nuosekliosios kinematinės grandinės robotams, apkrova darbiniam taške pasiskirsto tolygiai;
- lygiagrečios konstrukcijos standumas yra žymiai didesnis nei nuosekliosios.

Būtent dėl to teigiamos lygiagrečiųjų robotų savybės leidžia santykinai nedideliems lygiagretiesiems robotams manipuluoti didelėmis apkrovomis. Bėgant laikui dėl šių ir kitų priežasčių Stiuarto platformos pritaikymo galimybės prasiplėtė. Viena plačiausių pritaikymo sričių yra pozicionavimo sistemos. Pozicionavimą nanometrų skalėje leidžia atlikti tokios naujos technologijos kaip pjezoelektriniai vykdikliai, didelio tikslumo elektromechaniniai vykdikliai ar pneumatiniai-hidrauliniai vykdikliai su didelės skiriamosios gebos grįžtamojo ryšio jutikliais.

1.6.1. Tikslus pozicionavimas

Pavyzdžiui Rui Chen savo darbe pritaikęs Stiuarto platformą medicinoje sukūrė prietaisą, kuris gali pozicijuoti įrankį 10 μm tikslumu 10 mm spindulio darbinėje erdvėje, pačio įrankio apkrova 10N. Šiuo atveju naudojami elektromechaniniai servo vykdikliai su grįžtamoju ryšiu, o pats manipulatorius valdomas apibendrintųjų grandžių srityje naudojant atvirkštinį padėties uždavinį, bei įvertinant vykdiklių apkrovas.



7 pav. Elektromechaninis servo vykdiklis [13]

1.6.2. NASA LIDS (low impact docking system)

Amerikos Nacionalinės aeronautikos ir kosmoso administracijos sukurta kosminių sistemų švartavimo sistema. Manipulatoriaus geometrija artima 3-3 konfigūracijos Stiuarto platformai. Jis gali būti valdomas autonominiu arba rankiniu būdu. Šiuo atveju naudojami elektromechaniniai vykdikliai bei apkrovos jutikliai.



8 pav. NASA LIDS [14]

1.6.3. Bangų stabilizavimo sistema laivo kranui

Šiuo atveju stabilizuojamas laivo kranų judesys bei didelis išvystomos jėgos – manipulatoriaus masės santykis. Tikslumas ir greitis pritaikomi sklandžiam krovinių pakrovimui ar iškrovimui esant net ir dideliame bangavime (bangos aukščiui siekiant iki 4 m.).



9 pav. Laivo kranas jūroje [15]

1.6.4. Lygiagrečios kinematinės struktūros kranai (RoboCrane)

Apverstos Stiuarto platformos geometrijos manipulatoriai, kurių vykdikliai yra gervės su servo varikliais. Turi šešis laisvės laipsnius, leistina apkrova iki 855 kg, judėjimo greitis – 3 cm/s.

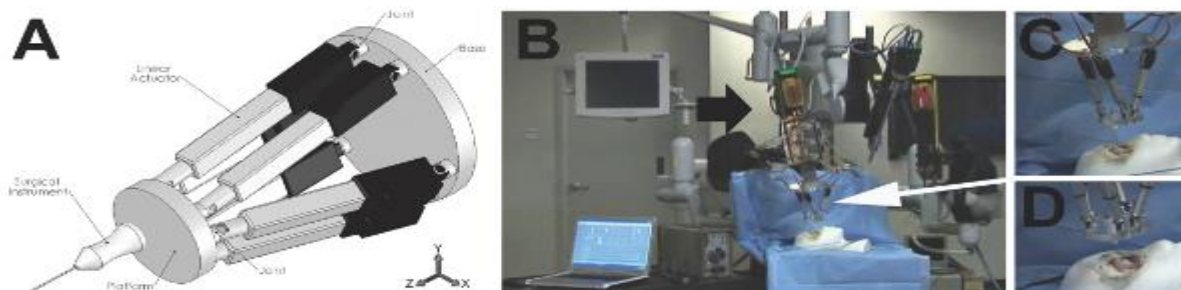


10 pav. „Gervė“ su servo varikliais [16]

1.6.5. Medicinos sritis

Panaudojant Stiuarto platformos modelį sukuriama įrenginiai, kurių dėka palengvinamas biomedicininis ir klinikinis žmogaus ištyrimas.

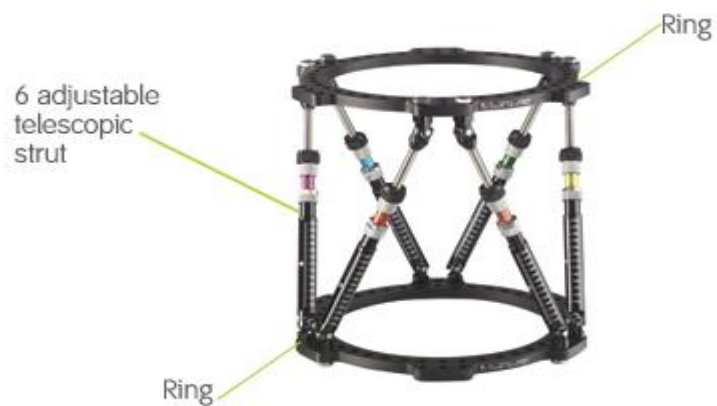
1.6.5.1 Oftalmologinėms operacijoms atlikti naudojami šiuo principu pagaminti prietaisai.



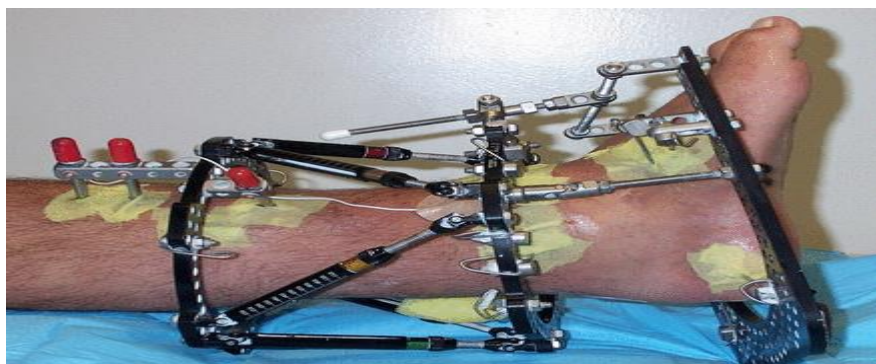
11 pav. Oftalmoskopas [17]

1.6.5.2. „Taylor Spatial Frame”

Gydytoja J. Charles Taylor sukūrė išorinei lūžių fiksacijai naudojamą prietaisą, kuris buvo pavadintas jos garbei.



12 pav. „Taylor Spatial Frame“ [18]



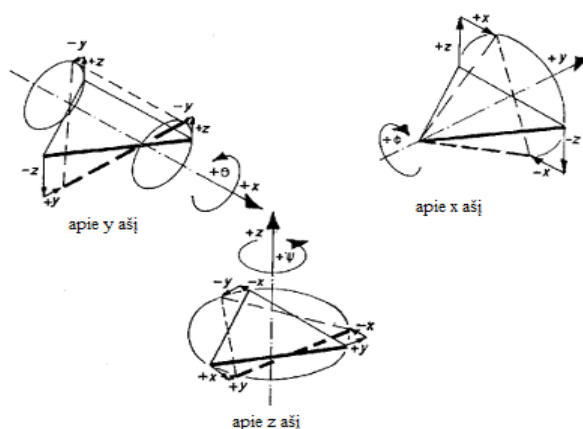
1.7. Skyriaus išvados

1. Stiuarto platforma – lygiagrečios kinematinės grandinės mechanizmas, jos padėtys aprašomos išsprendus tiesiogines ir atvirkštines kinematinės lygtis. Tiesioginės kinematikos – tai sudėtingos lygčių sistemos, kurios kai kuriais atvejais gali turėti labai daug realiųjų sprendinių (nuo 12 iki 40).
2. Stiuarto platformos valdymui dažniausiai naudojamos atvirkštinės kinematikos lygtys, kurios leidžia mechanizmą valdyti absoliučioje koordinačių sistemose.
3. Stiuarto platformos mechanizmo dinamiką aprašančios lygtys gali būti išvedamos keliais būdais: Lagranžo, Niutono-Oilerio ir Keino metodais. Tačiau, dinaminis mechanizmo modelis negali būti be greičių uždavinio sprendimo, naudojant Jakobianą, kurio skaičiavimas sudėtingas, todėl dėl šios priežasties gali apriboti tokio modulatoriaus panaudojimą valdymo sistemose.
4. Manipulatoriaus valdymo sistemoms dažniausiai naudojamos nesudėtingos grandžių apskaičiavimo lygtys, kai sprendžiami atvirkštinių padėčių ir greičių uždaviniai.

2. METODOLOGINĖ DALIS

2.1. Stiuarto platformos parametrų nustatymas

Stiuarto platformos kinematinis modelis sudaromas, naudojant atvirkštinės kinematikos lygtis. Šiuo modeliu apskaičiuojami vykdyklių kojų ilgiai, kuomet žinomi platformos padėčių taškai absoliučioje koordinatinių sistemje. Atvirkštinės kinematikos lygtys sudaromos keičiant manipuliatorių tam tikrose judėjimo trajektorijose, iš norimos padėties erdvėje apskaičiuojant užduotis tiriamam manipuliatoriui. Pati užduotis gali būti formuojama kaip padėties vektorius X , susidedantis iš 3 linijinių ir 3 ašinių poslinkių (1).



14 pav. Stiuarto platformos judėjimo trajektorijos x;y;z ašių atžvilgiu [13]

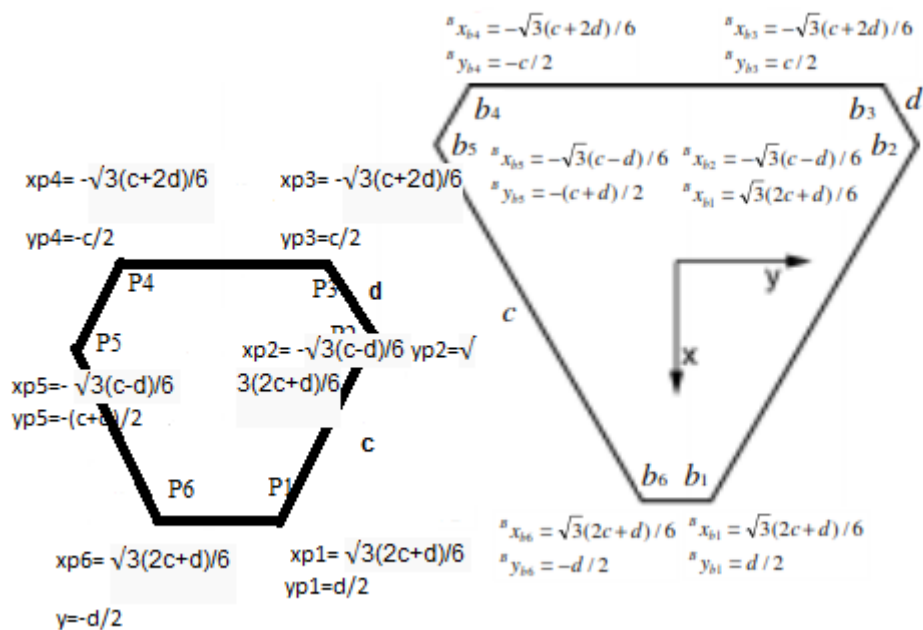
Pradiniai modelio parametrai: judančios platformos aukštis, judančios ir stabilios platformų kraštinių ilgių, atraminių taškų išdėstymo kampas $\alpha = 180^\circ$, atitraukimo kampai (angl. Offset) atraminiams taškams $\beta = [0^\circ ; 45^\circ]$, bei judančios platformos atraminių taškų pasukimo kampas γ , dažniausia 90° .

Judančios platformos atraminių taškų padėtis koordinatinių pradžios atžvilgiu apskaičiuojama pagal:

$$a_i = p_i * R + T, \quad (13)$$

čia : a_i – judančios platformos i-tasis atramos taškas, platformos koordinatinių sistemoje, p_i – i-tojo atraminio platformos taško vektorius absoliučioje koordinatinių sistemoje, R – Oilerio kampų posūkių matrica, T - perkėlimo vektorius.

Turint duomenis apskaičiuojamos visų taškų (Judraus pagrindo –P-P5 ir stabilaus pagrindo B1-B6) koordinatės XYZ koordinatinių sistemoje.



15 pav. Stabilios ir judrios platformų pradinių koordinatinių taškų apskaičiavimas

Platformos erdvinei orientacijai nusakyti naudojama 3x6 dydžio posūkių matrica R, sudaryta iš Oilerio posūkių kampų aplink O Z , O Y ir O X ašis atitinkamai (2), nurodanti manipulatoriaus judančios platformos koordinatinių sistemos posūkį.

$$R = \begin{bmatrix} \cos\alpha \cos\beta & \sin\alpha \sin\beta \sin\gamma - \sin\alpha \sin\gamma & \cos\alpha \sin\beta \cos\gamma + \sin\alpha \sin\gamma \\ \sin\alpha \cos\beta & \sin\alpha \sin\beta \sin\gamma + \cos\alpha \cos\gamma & \sin\alpha \sin\beta \cos\gamma - \cos\alpha \sin\gamma \\ -\sin\beta & \cos\beta \sin\gamma & \cos\beta \cos\gamma \end{bmatrix}, \quad (14)$$

Čia: $\sin(\alpha / \beta / \gamma)$, $\cos(\alpha / \beta / \gamma)$ - α, β, γ Oilerio posūkių kampai.

Kita labai svarbi matrica - padėties matrica T, ji nurodo judančios platformos koordinatinių sistemos pradžios taško O' padėtį absoliučioje koordinatinių sistemoje. Kitaip sakant , ši formulė padeda nustatyti abiejų platformų (judriosios ir stabiliosios) vidurio taškus.

$$T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}, \quad (15)$$

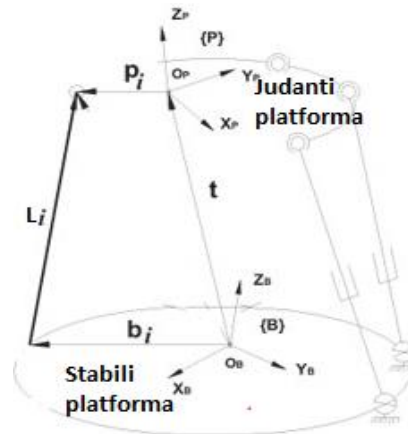
čia x, y ir z – taško O' koordinatės Dekarto koordinatinių sistemoje.

Žinant visus anksčiau paminėtus parametrus apskaičiuojami kojų ilgiai vektorinėje sistemoje:

$$L_i = R \cdot p_i + T - b_i (i = 1, \dots, 6), \quad (16)$$

$$[Li] = \begin{bmatrix} R \\ R \\ R \end{bmatrix} \cdot \begin{bmatrix} xp \\ yp \\ zp \end{bmatrix} + \begin{bmatrix} T \\ T \\ T \end{bmatrix} - \begin{bmatrix} xb \\ yb \\ zb \end{bmatrix}, \quad (17)$$

čia: L – kojos ilgis, išreikštas vektoriais; i – kojos numeris (1,...,6), R - judėjimo matrica p-viršutinės (judančios) platformos kojų koordinatės, T - padėties koordinatės, bi- apatinės/ stabiliosios platformos kojų taškų koordinatės.



16 pav Vektoriai , naudojami atvirkštiniam padėčių uždaviniui spręsti

Naudojantis aprašytomis lygtimis, MATLAB programavimo aplinkoje sukuriama programa, apskaičiuojanti judesio modeliavimui reikalingus pradinius geometrinius parametrus pagal pageidaujamo vektoriaus X, Y, Z duomenis (trajektoriją). Funkcijos kodas pateiktas 1 priede.

Pradiniame modelio sudarymo ir tyrimo etape nėra žinomos ribos, kuriose pati platforma juda, nėra žinomas maksimalus kojų ilgis koordinatinių sistemoje, bei taip nežinomas maksimalus ir minimalus galimas jo posūkio kampas X, Y, Z ašyse α, β, γ posūkio kampų atžvilgiu.

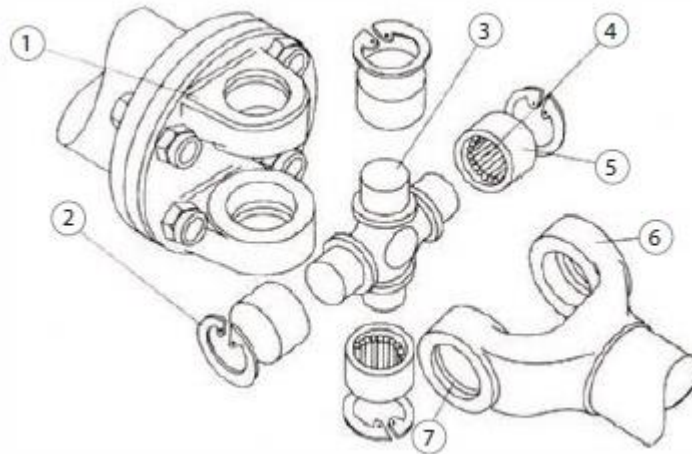
Funkcija parašyta laikant, jog pradinėje padėtyje analizuojamo manipulatoriaus judančios platformos centras yra toje pačioje tiesėje, kaip ir pagrindo platformos bei abiejų platformų z ašys ir pati platforma nepasukta erdvėje pagrindo atžvilgiu, todėl posūkio matrica R yra lygi vienetinei 3×3 dydžio matricai.

Analizuojamo manipulatoriaus dinamika buvo modeliuojama naudojantis MATLAB Simulink SimMechanics funprogramos paketu. Prieš naudojant šios programos paketą būtina rasti tam tikrus parametrus:

- pagrindo ir judriosios platformos sąnarų posūkių ašis;
- tiriamo manipulatoriaus visų 6 kojų ilgius bei jų vektorius;
- hidraulinio cilindro ir koto l ilgių vektorius, gravitacijos centruose vektorius, posūkio vektorius bei judančių dalių mases, jėgas.

Manipulatoriaus dinaminis modelis SimMechanics programoje ir parametų lygčių sudarymas ir apskaičiavimas buvo atliktas remiantis straipsniu [9]. Universalųjų sąnarų ašių ortai apskaičiuojami kiekvienos kojos dviejuose skirtinguose taškuose:

1. B_i – pagrindo (stabiliojo) atraminiam taške;
2. P_i – platformos (judriosios) atraminiam taške.



17pav. Universaliosios sąnaros principinė schema; 1 - šakutė, 2 - Seeger fiksuojantis žiedas, 3 - kryžmė, 4 - adatinis guolis, 5 - adatinio guolio įvorė, 6 - Seeger žiedo griovelis [14]

Universalioji, dar kitaip vadinama kardaninė sąnara, turi du sukimosi laisvės laipsnius kampus, kurie statmeni viena skitam (4pav). Sąnaros 1-osios ašies ortas lygus sąnaros sujungtos kojos vektoriaus ir Dekarto kordinačių sistemoje Z ašies vektoriaus sandaugai, todėl gautasis rezultatas statmenas ašių sudaromai plokštumai. 2-osios ašies ortas lygus iš 1-osios ašies orto ir su sąnara sujungtos kojos vektorių sandaugai- šiuo atveju vėlgi gautasis 2-osios ašies ortas yra statmenas 1-osios ašies ortui ir su sąnara sujungtos kojos vektoriui. Universalųjų sąnarų taškuose B_i ir P_i ortai tokie patys.

Analizuojamo manipulatoriaus kojų ilgių vektoriai apskaičiuojami pagal jau minąją lygtį, paremtą atvirkštine kinematinė lygtimi, kuomet žinomi judančios ir atraminės platformos taškų koordinatės ir vektoriai. MATLAB funkcijos kodas skirtas kojų parametrus rasti pateiktas 2 priede.

Norint apskaičiuoti turimo cilindro korpuso ir koto parametrus (gravitacijos centrus, mases, jėgas ir kt.), nominalus kojos vektoriaus ilgis dalinamas į santykinius ilgius (5pav.). Numatytieji kojos ilgio vektoriaus santykiai: cilindro ilgio vektoriui - 3/4 kojos ilgio vektoriaus ir koto ilgio vektoriui - 1/4 kojos ilgio vektoriaus.

Panašiai apskaičiuojami ir gravitacijos centrų vektoriai. Cilindro korpuso gravitacijos centro vektorius atitinka 3/8 manipulatoriaus kojos ilgio vektoriaus dalį, o koto gravitacijos centras – 5/8 kojos ilgio vektoriaus.

Dinaminio Stiuarto platformos modelio judančių dalių jėgos, ilgių vektoriai apskaičiuojami naudojantis MATLAB programo paketu.

2.2. Stiuarto platformos vykdyklių modeliavimas

Hidraulinės servo sistemos, kuriomis tiriamas manipulatorius valdomas, pasižymi geromis išvystomomis jėgos ir sistemos masės santykiu, bei tikslumo charakteristikomis, tačiau šie turimi rodmenys dažnai pranašesni nei kitų mechaninių ir pneumatinių valdymo sistemų charakteristikos. Matematinis šios sistemos modelis sudaromas iš :

- hidraulinio cilindro;
- servo vožtuvo lygčių sistemų (kurios susietos esamo hidraulinio skysčio debitais Q_1 ir Q_2).

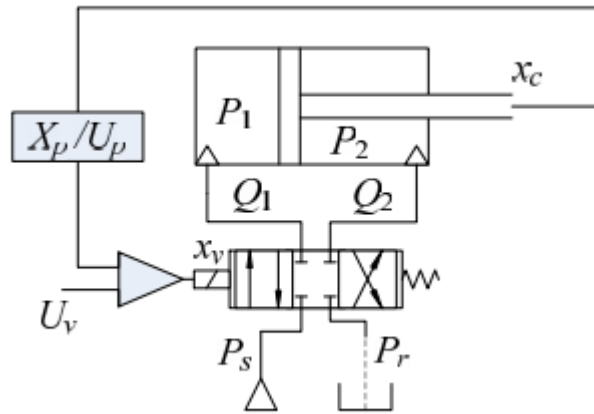
[7]

Šių debitų bei slėgių dydžiai skiriasi nuo to kurioje jos kameros pusėje yra, bei nuo to į kurią pusę juda cilindro kotas. Debitai išreiškiami lygčių sistemomis (18,19). [10]:

$$Q_1 = \begin{cases} C_q w x_v \sqrt{\frac{2}{\rho}} (p_s - p_1), x_v \geq 0; \\ C_q w x_v \sqrt{\frac{2}{\rho}} (p_1 - p_r), x_v < 0 \end{cases}, \quad (18)$$

$$Q_2 = \begin{cases} C_q w x_v \sqrt{\frac{2}{\rho}} (p_2 - p_r), x_v \geq 0; \\ C_q w x_v \sqrt{\frac{2}{\rho}} (p_s - p_2), x_v < 0 \end{cases}, \quad (19)$$

čia: C_q – debito koeficientas vožtuve; w – vožtuvo ploto gradientas, m; x_v – vožtuvo sklendės padėtis, m; ρ – hidraulinio skysčio tankis, m^3/kg ; p_1 ir p_2 – atitinkamai slėgiai cilindro kameroje, Pa; p_s – darbinis slėgis, Pa; p_r – skysčio rezervuaro slėgis, Pa.



18 pav. Hidraulinės servo pavaros principinė schema

Patį cilindrą veikiančios jėgos, sukeltos skysčio slėgio, aprašomos hidraulinio skysčio slėgio lygtimis [14]:

$$\frac{dp_1}{dt} = \frac{\beta_s}{V_1} = (Q_1 - A_1 v), \quad (20)$$

$$\frac{dp_2}{dt} = \frac{\beta_s}{V_2} = (A_2 v - Q_2), \quad (21)$$

čia: β_s – skysčio spūdumo modulis, Pa; V_1 ir V_2 – cilindro kamerų tūriai, m^3 ; A_1 ir A_2 – stūmoklio darbiniai plotai, m^2 ; v – koto greitis, m/s.

Cilindro koto judesio lygtis (22) sudaroma ir apskaičiuojama pasinaudojus Niutono 2-uoju dėsniumi [11]:

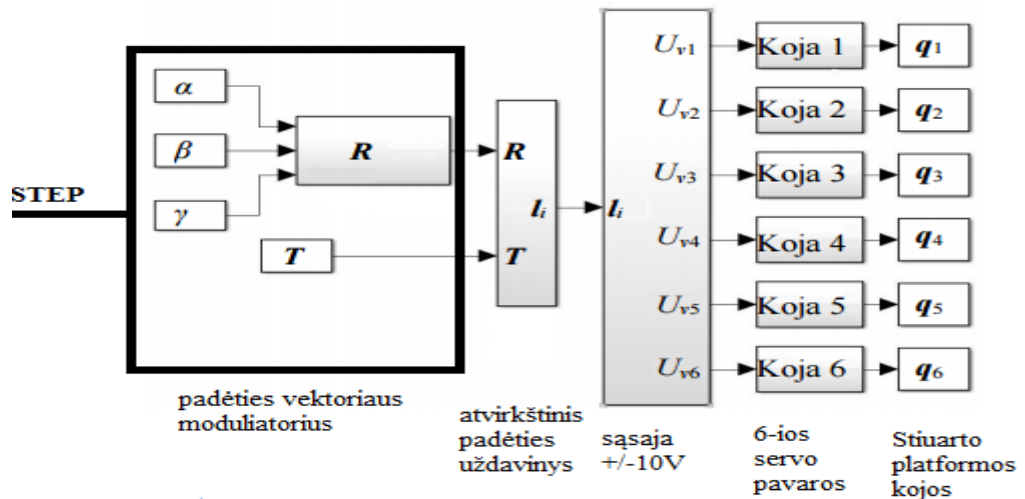
$$\frac{dv}{dt} = p_1 A_1 - p_2 A_2 - F_a, \quad (22)$$

čia: F_a – cilindro apkrovos jėga, N.

Apskaičiavus visas prieš tai aptartas lygčių sistemas, toliau duomenis suvedami į SimuLink programos paketą ir taip suformuojamas Stiuarto platformos mechaninės dalies modelis (5pav).

2.3. Stiuarto platformos valdymo modeliavimas Simulink programoje

Stiuarto platformos valdymo sistema modeliuojama Simulink ir Matlab programų paketu terpėse. Visos prieš tai minėtos hidraulinės sistemos matematiniai modeliai perkeliama į Simulink programos vadinamąjį „bloko“ modelį (19pav). Tart būsimo trajektorijos keitimo bloko ir servo pavaros įterpta sąsaja, kitaip dar vadinamas keitimo blokas, +/- 10V , kuris generuoja analoginio valdymo signalo komandą.



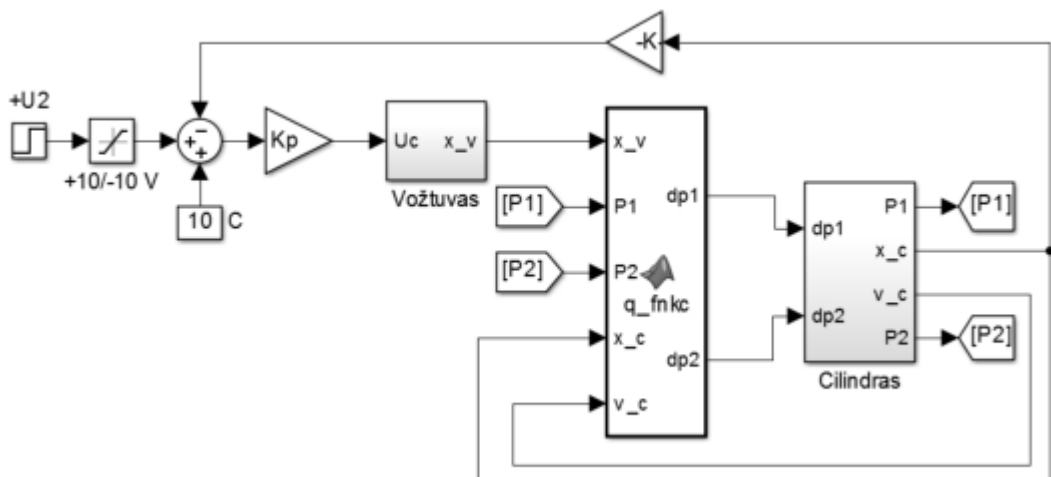
19 pav. Stiuarto platformos valdymo grandžių modelis Simulink pogramoje

Būtent reikalingo hidraulinio servo pavaros vykdiklio dinaminiam modeliui sudaryti panaudotos anksčiau pateiktos lygtys (18;19). Kadangi pats sudarytas modelis netiesinis, todėl simuliacijos parametruose buvo stengiamasi nustatyti nekintančio žingsnio diferencialinių lygčių sprendimo algoritmas. Kuriant modelį buvo panaudoti 1 lentelėje pateikti duomenys.

1 lentelė

Parametrų reikšmės

<i>Parametras</i>	<i>Vertė</i>	<i>Reikšmė</i>
P1	110mm ²	Pilnas stūmoklio plotas
P2	21mm ²	Kojos pusės plotas
l	210mm	Kojos ilgis
P _s	160bar	Sistemos darbinis slėgis
B _s	6890bar	Skysčio spūdomo modelis
W	2,5mm	Sklendės ploto gradientas
C _q	0,61	Debito koeficientas
P	890 kg/m ³	Hidraulinio skysčio tankis



20 pav. Hidraulinės servo pavaros matematinis analogas Simulink programos pakete

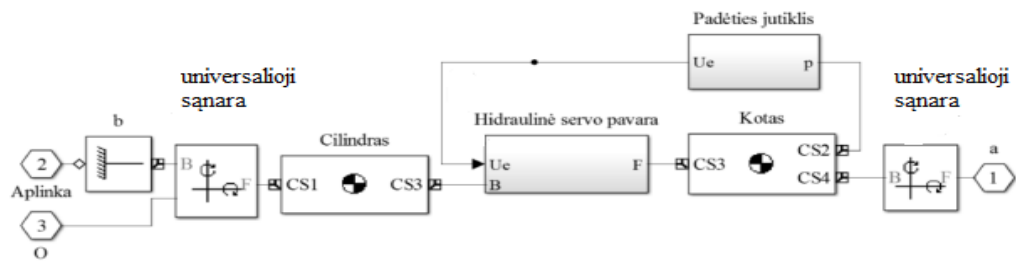
2.4. Stiuarto platformos modelis Simulink programoje

Panaudojus 2.1-2.3 skyriuose minėtomis formulėmis, kurios apibūdina analizuojamo objekto kinematiką ir dinematiką matematinėmis išraiškomis, toliau buvo kuriamas Simulink „blokų“ modelis, kuri panaudotas pačio judesio modeliavimui.

SimMechanics funkcinių blokų biblioteka panaudojama būtent tokių stambių kūnų modeliavimu trimatėje erdvėje. Du pagrindiniai blokai, kurie naudojami: mechaniniai kūnai ir sąnarus. Panaudoti funkciniai blokai apibūdina šiuos parametrus:

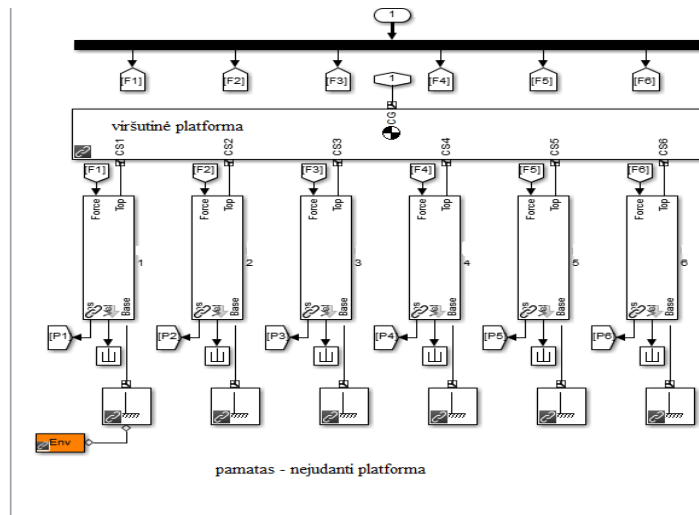
- Kūno masę, kg;
- Kūno inercijos momentą, $\text{kg}\cdot\text{m}^2$;
- Kūno gravitacijos centrą, $[X,Y,Z]$;
- Kūno taškus x,y,z koordinačių sistemoje, bei 3×6 posūkio matricioje.

Stiuarto platformos modelio kojos mechanizmas sudaromas iš 2 dalių: cilindro korpuso ir koto bei 3 sąnarių. 2 iš 3 sąnarių – tai universaliosios, kurios jungia cilindrą su pagrindu ir judančia platforma. Visos šios dalys standžios, nekintančios. Kojos modelis papildomas hidrauline servo pvara, kuri įsiterpia tarp cilindro ir koto. (21 pav.).



21 pav. Kojos modelis Simulink programoje

Taip sujungus visas šešias platformos kojas, gaunama blokų sistema – pilnas Stiuarto platformos mechanizmo modelis (22 pav.). Taikoma, kad sistemoje modeliuojamas laisvo kritimo pagreitis, veikiant neigiamos y ašies kryptimi - todėl nurodomas laisvo kritimo pagreitis vektorinėje sistemoje – $[0, 0, -20]$.



22 pav. Stiuarto platformos mechaninės dalies modelis SimuLink programoje

2.5. Stiuarto platformos dinaminių parametrų radimas

Sudarytam Stiuarto platformos modeliui toliau reikia apskaičiuoti pereinamojo procesos parametrus – tai būtų:

- Stirpinimo koeficientas, K_{pr} ;
- Laiko pastovioji, T_0 ;
- Vėlavimo trukmė, τ_0 .

Laiko pastoviosios ir vėlavimo trukmės apskaičiavimas gali būt atliekamas dviem būdais:

- Apskaičiuojant liestinę proceso kreivės perlinkimo taške;
- Ir ieškant tiesės, einančios per du proceso taškus esančius 30% ir 60% nusistovėjusio pereinamojo proceso kreivės skaitinės vertės .

Modeliuojamos Stiuarto platformos stiprinimo koeficientas randamas iš pereinamojo proceso nusistovėjusios vertės ir valdomo objekto įėjimo dydžio pokyčio santykio

2.6. PID reguliatoriaus parametrų nustatymas

Kaip jau anksčiau buvo minėta, analizuojamo modelio vykdikliai – hidraulinės servo pavaros valdomos PID reguliatoriais, todėl jų reikiami parametrai nustatomi naudojantis literatūroje pateikta metodine medžiaga ir pateiktomis derinimo lentelėmis [12].

- Ziegler ir Nichols šuolinės reakcijos metodas, šiuo atveju naudojami du proceso dinamiką apibūdinantys parametrai (stiprinimas a ir trukmė τ_0);
- Chien, Hrodes ir Reswick metodas. Jo metu naudojami 3 dinaminiai parametrai – K_{pr} , τ_{pr} ir T_0 ;
- Cohen ir Coon metodas, derinimo taisyklės tokios pat kaip ir naudojant Chien, Hrodes ir Reswick metodą.

2.7. Skyriaus išvados

1. Stiuarto platformos kinematinis modelis sudaromas, naudojant atvirkštinės kinematikos lygtis X,Y,Z koordinačių sistemoje.

2. Pradiniai modelio parametrai: judančios platformos aukštis, judančios ir stabilios platformų kraštinių ilgiai, atraminių taškų išdėstymo kampas $\alpha = 180^\circ$, atitraukimo kampai (angl. Offset) atraminiams taškams $\beta = [0^\circ ; 45^\circ]$, bei judančios platformos atraminių taškų pasukimo kampas γ , dažniausia 90° .

3. Turint duomenis apskaičiuojamos visų taškų (Judraus pagrindo –P, P1, P2, P3, P4, P5 ir stabiliaus pagrindo B1-B6) koordinatės XYZ koordinačių sistemoje.

4. Platformos erdvinei orientacijai nusakyti naudojama 3x6 dydžio posūkio matrica R, sudaryta iš Oilerio posūkio kampų aplink O Z , O Y ir O X ašis atitinkamai (2), nurodanti manipulatoriaus judančios platformos koordinačių sistemos posūkį.

5. Padėties matrica T, ji nurodo judančios platformos koordinačių sistemos pradžios taško O' padėtį absoliučioje koordinačių sistemoje.

6. Universaliųjų sąnarų ašių ortai apskaičiuojami kiekvienos kojos dviejuose skirtinguose taškuose: B_i ir P_i –taškuose. Apskaičiavus ortus jie yra identiški.

7. Hidraulinės servo sistemos, kuriomis tiriamas manipulatorius valdomas, pasižymi geromis išvystomomis jėgos ir sistemos masės santykio, bei tikslumo charakteristikomis, tačiau šie

turimi rodmenys dažnai prasnesni nei kitų mechaninių ir pneumatinių valdymo sistemų charakteristikos.

8. Naudojant SimuLink programos paketą sufomuojamas Stiuarto platformos mechaninės dalties modelis.

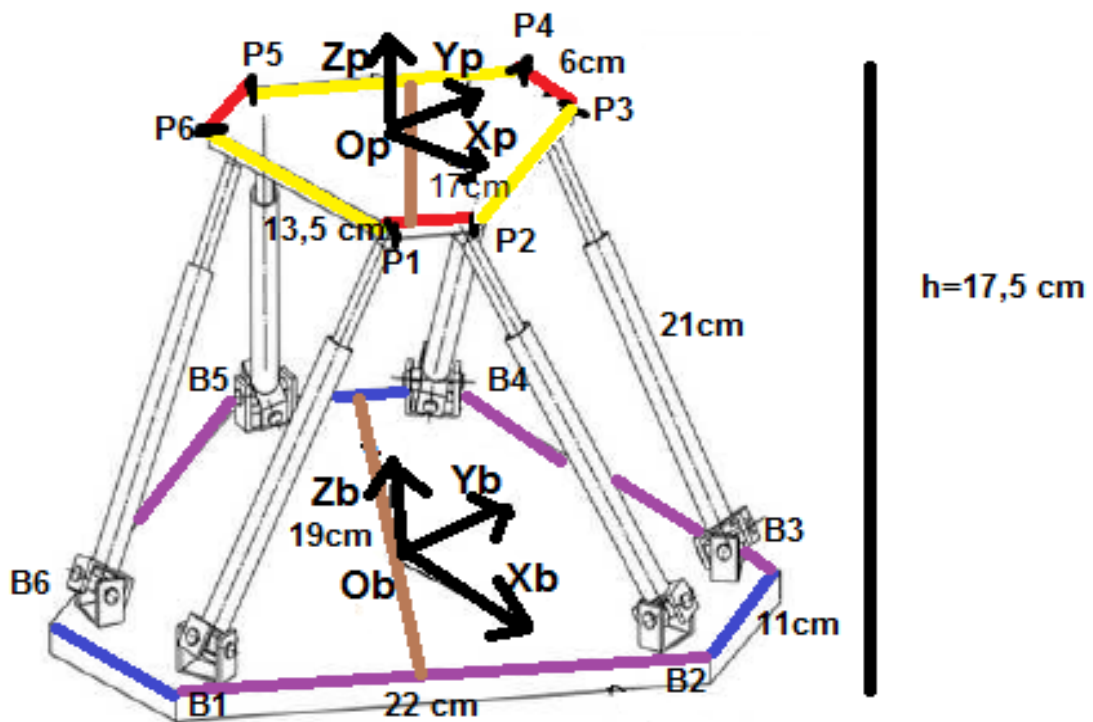
9. PID regulatoriaus parametria gali būti apskaičiuojami remiantis: Ziegler ir Nichols šuolinės reakcijos , Chien, Hrodes ir Reswick arba Cohen ir Coon metodu.

10. Stiuarto platformos valdymo sistema modeliuojama Simulink ir Matlab programų paketų terpėse. Visos hidraulinės sistemos matematiniai modeliai perkeliama į Simulink programos vadinamąjį „bloko“ modelį.

11. Stiuarto platformos modelio kojos mechanizmas sudaromas iš 2 dalių: cilindro korpuso ir koto bei 3 sąnarų. 2 iš 3 sąnarų – tai universaliosios, kurios jungia cilindrą su pagrindu ir judančia paltforma.

3. TIRIAMOJI DALIS

Parinkta geometrinė konstrukcija Stiuarto platformai, kurios duomenys bus toliau naudojami sudarant , sprendžiant lygtis ir toliau modeliuojant MatLab programos pakete. Stiuarto platformos koncepcija pavaizduota 22 pav.



22 pav. Stiuarto platformos koncepcija

Ieškant optimaliausių geometrinės Stiuarto platformos konfigūracijų, naudojami šie parametrai:

1. linijiniai poslinkiai poslinkiai:

- vertikalus linijinis poslinkis (z ašies kryptimi): ± 200 mm.
- horizontalūs linijiniai poslinkiai (x ir y ašių kryptimis): ± 100 mm.

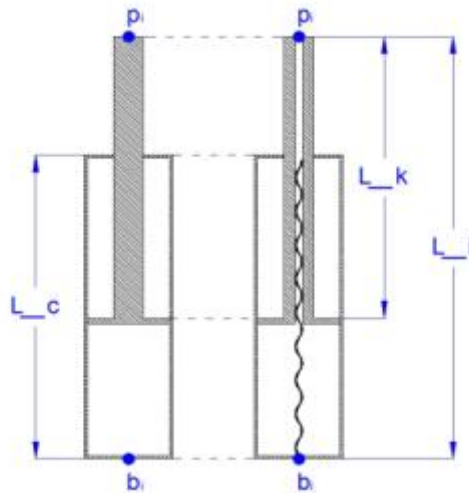
2. ašiniai posūkiai:

- posūkis apie z ašį: $\pm 45^\circ$.
- posūkis apie y ašį: $\pm 45^\circ$.
- posūkis apie x ašį: $\pm 45^\circ$.

3. pagrindo ir judančios platformos apskritimų spinduliai, kurie bus naudojami skaičiavimuose:

- pagrindo apskritimo spindulys: 190 mm.
- judančios platformos apskritimo spindulys: 170 mm.

Pagal turimus parametrus toliau bus skaičiuojami kojų ilgiai laikant, kad kojos – tai hidraulinių cilindrų modeliai, kuriems galioja paprasti jų apribojimų elementai. (24 pav.)



24 pav. Manipulatoriaus kojų modelis

Taip pat, naudojant esamus parametrus apskaičiuojamos pradinės platformos koordinatės, kojų ilgiai – kurie vėliau panaudojami platformos valdymo tyrime pačioje MatLab programos pakete.

3.1. Manipulatoriaus valdymas naudojant Simulink programos paketą

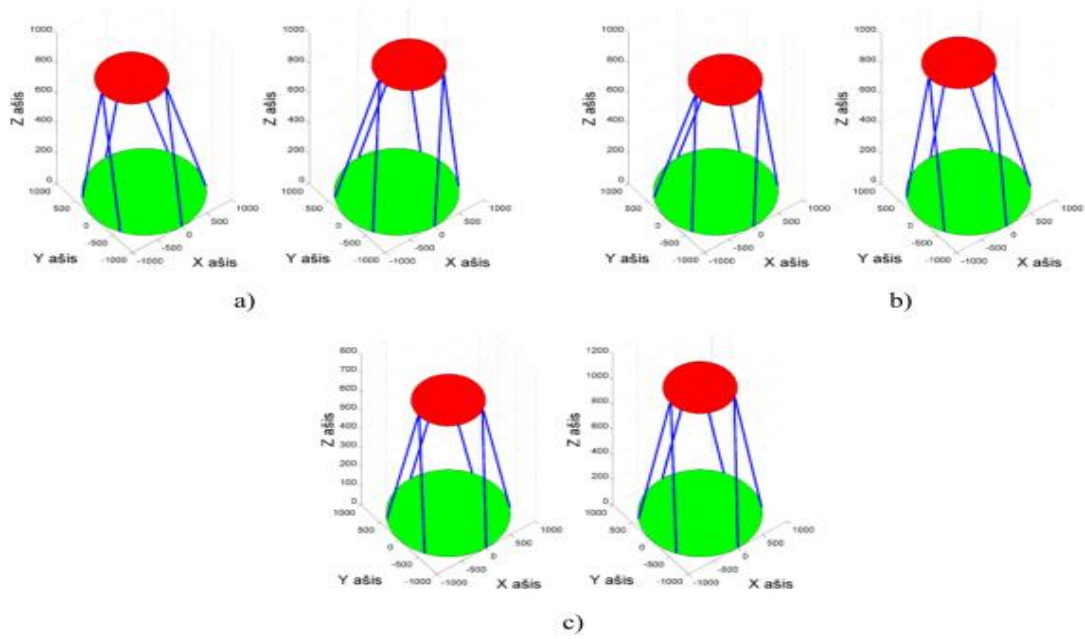
Sukurtas Stiuarto platformos modelis, kaip kad buvo aptarta metodologinėje dalyje, Simulink programos pakete toliau tikrinamas 3 skirtingų konfigūracijų aspektu: 3-3, 3-6, 6-6. Atliekami skaičiavimai ir gauti duomenys: vykdyklių ilgių rezultatai visais trim pasirinktais atvejais turėtų išspausdinti į txt failą. Visoms konfigūracijoms parinkti vykdykliai netiko, sistema rodė klaidą ir norimų duomenų nebuvo gauta. Vėliau bandymas buvo atliktas su kitų parinktų ilgių vykdykliais. Platforma judėjo 6 skirtingomis kryptimis, todėl ir duomenys buvo ieškomi 6 skirtingais aspektais.

Judėjimas linijiniais poslinkiais:

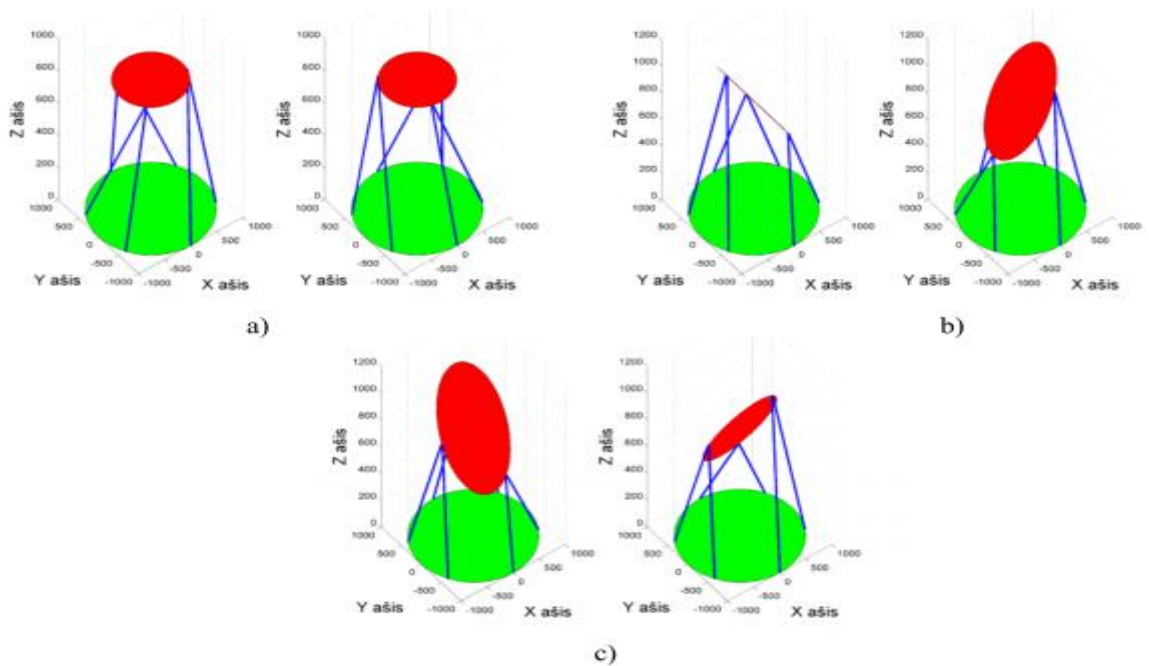
- X ašies kryptimi (25 pav. a)
- Y ašies kryptimi (25 pav. b)
- Z ašies kryptimi (25 pav. c)

Judėjimas ašiniiais poslinkiais:

- α apie Z ašį (26 pav. a)
- β apie Y ašį (26 pav. b)
- γ apie X ašį (26 pav. c)

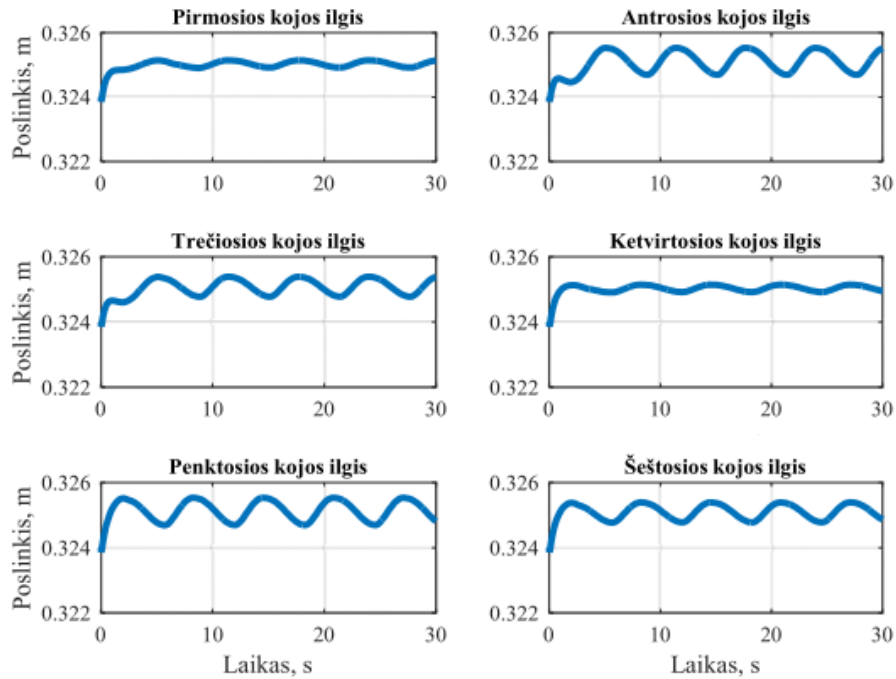


25 pav. Manipulatoriaus linijiniai poslinkiai



26 pav. Manipulatoriaus ašiniai poslinkiai

Sudarytas vykdyklio modelis išbandomas ir paleidžiamas per naudojamą Simulink programą, jo valdymo sistemai užduodant 2 priešingus valdymo tikslus – atitinkamai X, Y ašimis. Vykdyklio ilgio kitimas vertinant gautą modelį, turėtų atitikti apvalią roboto centrinio tipo trajektoriją (27 pav.)

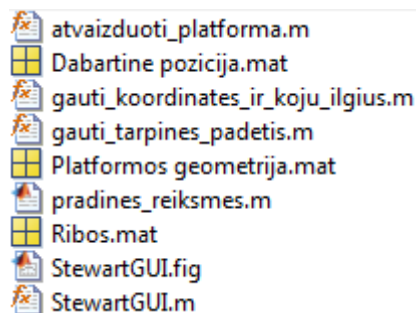


27 pav. Manipulatoriaus vykdyklių kitimo trajektorijos per tam tikrą laiką X ir Y ašyse

Tačiau rezultatai nėra tikslūs, kadangi paleidžiama programa stringa, kiekvieno skaičiavimo metu naudojant tuos pačius parametrus gaunami vis skirtingi duomenys. Todėl didesnis dėmesys buvo skiriamas MatLab programavimui ir Stiuarto platformos erdvinio kūno kūrimui bei jo modeliavimui.

3.2. Matlab rogramos struktūros apibendrinimas

Tam, kad būtų lengviau dirbti ir orientuotis, programa buvo padalinta į atskirus failus. Programa susideda iš 5 matlab kodo failų, grafinio vartotojo sąsajos lango ir 3 duomenų failų.



28 pav. Matlab programos failų turinys

Programiniai „m“ failai:

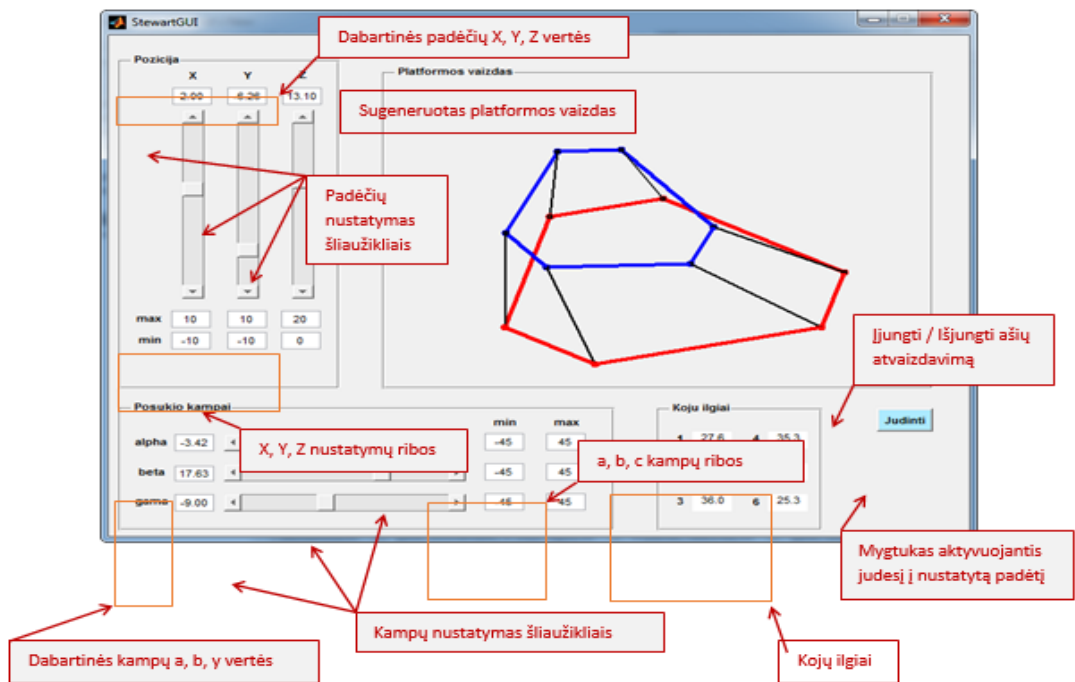
- **pradines_reiksmes.m** – šis failas turi būti paleidžiamas prieš naudojant „StewartGUI.m“ failą, kadangi jis sugeneruoja pradinis geometrijos taškus. Šį failą pakanka paleisti vieną kartą.
- **atvaizduoti_platforma.m** – platforma atvaizduojama lange.
- **gauti_koordinates_ir_koju_ilgius.m** – funkcija apskaičiuojanti platformos koordinates pagal įvestas pozicijos ir posūkio vertes;
- **gauti_tarpines_padetis** – judesio dinamikos atvaizdavimui skirta funkcija. Ji sugeneruoja tarpinius taškus nuo pradinės padėties iki pasirinktos padėties. Atvaizduojant tarpinius taškus, matomas platformos judėjimas;
- **StewartGUI.m** – grafinės vartotojo sąsajos programinis kodas;

Duomenų failai:

- **Platformos geometrija.mat** - ;
- **Dabartine pozicija.mat** – išsaugoma dabartinė platformos pozicija;
- **Ribos.mat** – šiame faile saugomos vertės, nustatančios judesio X, Y, Z, bei posūkio kampų a, b, c ribas;

Grafinės vartotojo sąsajos langas:

StewartGUI.fig – juo valdoma platformos padėtis ir atvaizduojamas platformos vaizdas, bei kojų ilgiai.

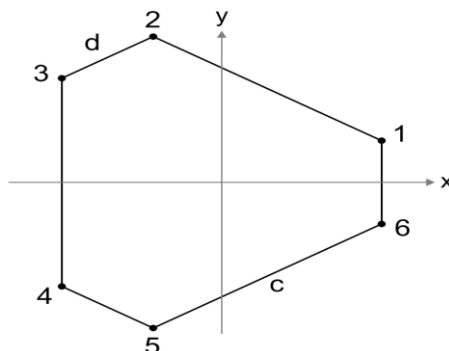


29 pav. Grafinės vartotojo sąsajos langas

3.3. Programinis kodas

3.3.1. Pradinių reikšmių apskaičiavimas

Paleidus šį failą sugeneruojami bazės ir judančios platformos geometrijos duomenys ir pradiniai padėties duomenys. Užduoties duomenys yra d ir c ilgiai bazės ir platformos geometrijai apibrėžti.



30 pav. Šešiakampo geometrija

2 lentelė.

Pradinės geometrijos skaičiavimas

Taško nr.	x koordinatė	y koordinatė	z koordinatė
1	$\sqrt{3}(2c + d)/6$	$d/2$	0
2	$-\sqrt{3}(c - d)/6$	$(c + d)/2$	0
3	$-\sqrt{3}(c + 2d)/6$	$c/2$	0
4	$-\sqrt{3}(c + 2d)/6$	$-c/2$	0
5	$-\sqrt{3}(c - d)/6$	$-(c + d)/2$	0
6	$\sqrt{3}(2c + d)/6$	$-d/2$	0

Šiomis formulėmis apskaičiuojamos platformos bazės ir judančios dalies koordinatės. (1 priedas)

3 lentelė

Bazės taškų koordinatės

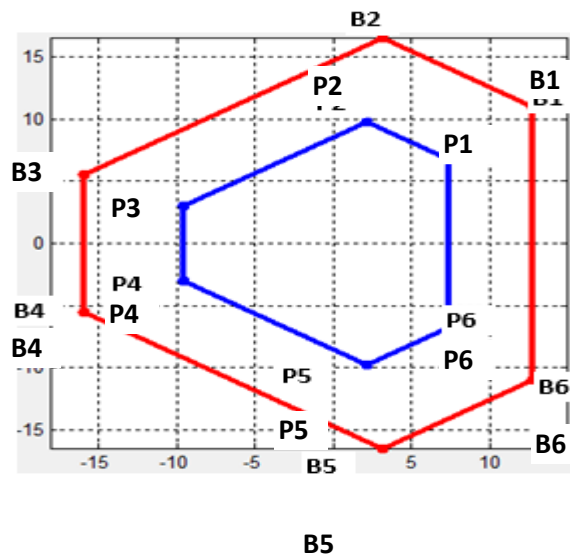
	B1	B2	B3	B4	B5	B6
x	12.702	3.175	-15.877	-15.877	3.175	12.702
y	11.000	16.504	5.500	-5.500	-16.500	-11.000
z	0	0	0	0	0	0

4 lentelė

Judančios platformos taškų koordinatės

	P1	P2	P3	P4	P5	P6
x	7.361	2.165	-9.526	-9.526	2.165	7.361
y	6.750	9.750	3.000	-3.000	-9.750	-6.750
z	0.000	0.000	0.000	0.000	0.000	0.000

B2



31 pav. Pradinė platformos geometrija (mėlyna - judanti dalis, raudona - bazė)

Gautos vertės išsaugomos faile “Platformos geometrija”, kad vėliau kitos programos funkcijos jas galėtų panaudoti. Taip pat šis failas sugeneruoja pradines ribas X,Y,Z ir a,b,y padėties ir posūkio kampų nustatymams, taip pat ir kojų spalvas.

3.3.2. Koordinatinių ir kojų ilgių apskaičiavimas, bei jų kitimas laike

Platformos taškų koordinatėms ir kojų ilgiams apskaičiuoti sukurama atskira funkcija ir išsaugoma atskirame .m faile (gauti_koordinates_ir_koju_ilgius.m).

Funkcija naudojama koordinatėms ir kojų ilgiams apskaičiuoti:

$$[B,P,L] = \text{gauti_koordinates_ir_koju_ilgius} (X,Y,Z,\alpha,\beta,\gamma)$$

Koordinatės apskaičiuojamos remiantis jau anksčiau aptartomis posūkio **R** ir padėties **T** matricomis.

$$R = \begin{bmatrix} c\alpha \cdot c\beta & s\alpha \cdot s\beta \cdot s\gamma - s\alpha \cdot s\gamma & c\alpha \cdot s\beta \cdot c\gamma + s\alpha \cdot s\gamma \\ s\alpha \cdot c\beta & s\alpha \cdot s\beta \cdot s\gamma + c\alpha \cdot c\gamma & s\alpha \cdot s\beta \cdot c\gamma - c\alpha \cdot s\gamma \\ -s\beta & c\beta \cdot s\gamma & c\beta \cdot c\gamma \end{bmatrix}, \quad (23)$$

$$T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}, \quad (24)$$

Judančios platformos koordinatės pagal X,Y,Z padėtį ir α,β,γ posūkio kampus:

$$P = R(\alpha,\beta,\gamma) \times P + T(X,Y,Z), \quad (25)$$

Kojų ilgis apskaičiuojamas radus koordinačių skirtumą tarp bazės ir judančios platformos taškų.

Pirmiausia randama kojų ilgių x,y,z koordinačių matrica:

$$Lm = R(\alpha, \beta, \gamma) \times P + T(X, Y, Z) - B, \quad (26)$$

Po to apskaičiuojami vektorių ilgiai:

$$L = \sqrt{Lm_x^2 + Lm_y^2 + Lm_z^2}, \quad (27)$$

Ir galiausiai apskaičiuojamos tarpinės padėties dančios platformos, kad būtų galima stebėti platformos judesį tarp buvusios padėties ir nustatytos. (3 priedas)

3.4. Kojų ilgių kitimo analizė

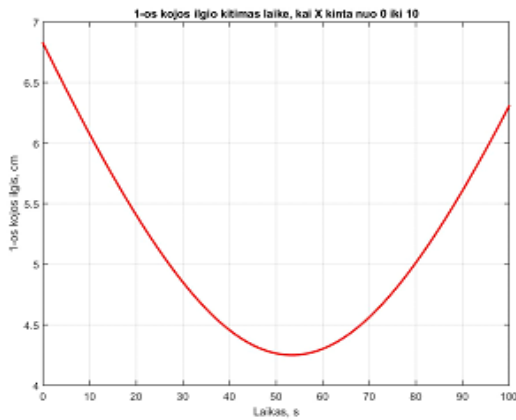
Kojų ilgių kitimo analizė atliekama atsižvelgiant į min. / maks. ribas X, Y, Z ašyje ir α, β, γ judėjimo kampus per tam tikrą laiką

Skaičiavimai pateikti lentelėse, spalvine gama nurodomas didžiausias ir mažiausias kojų ilgis (geltona – mažiausias, raudona – ilgiausias), bei grafiku vizualiai pavaizduotas kojų ilgio kitimas kintant X, Y, Z padėtimis ir kintant α, β, γ judėjimo kampams.

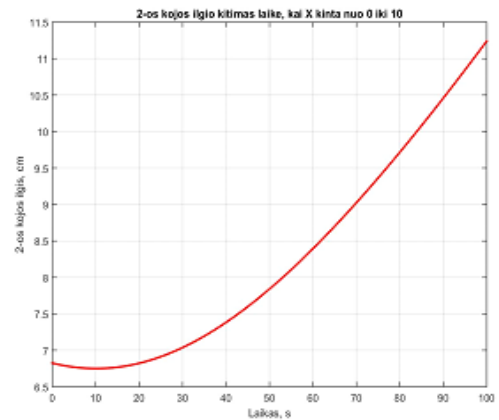
5 lentelė

Kojų ilgio priklausomybė nuo X padėties kitimo

X padėtis	Kojų ilgis					
	1	2	3	4	5	6
1	6.074	6.756	7.764	7.764	6.750	6.074
4	4.456	7.382	10.648	10.648	7.382	4.456
5	4.2636	7.840	11.622	11.622	7.840	4.263
8	5.013	9.716	14.567	14.567	9.711	5.013
10	6.306	11.247	16.549	16.549	11.241	6.036

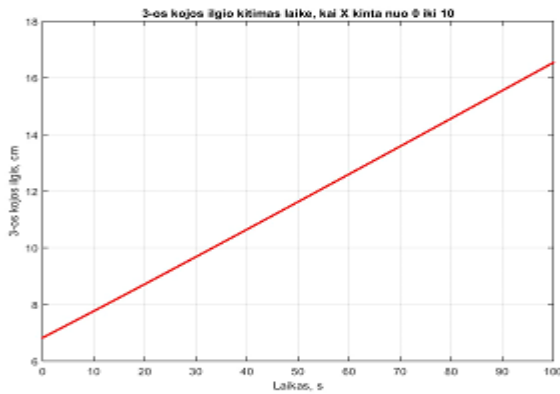


a)

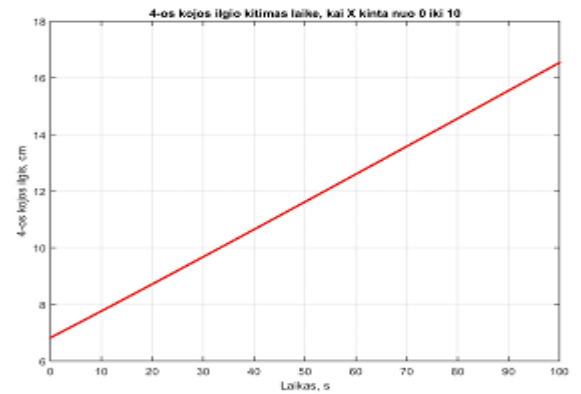


b)

32 pav. Kojų ilgių priklausomybė kintant X ašis kryptimi per tam tikrą laiką, a - 1 koja, b - 2 koja

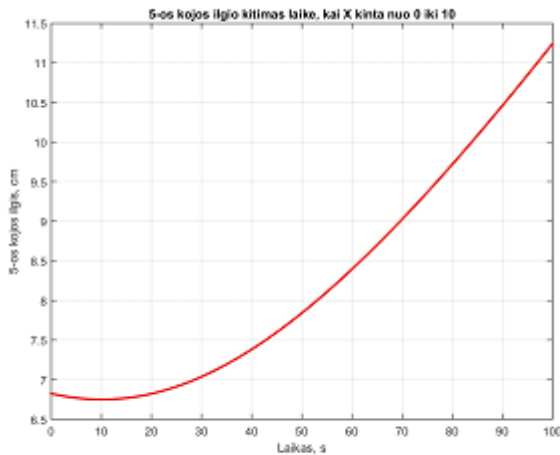


a)

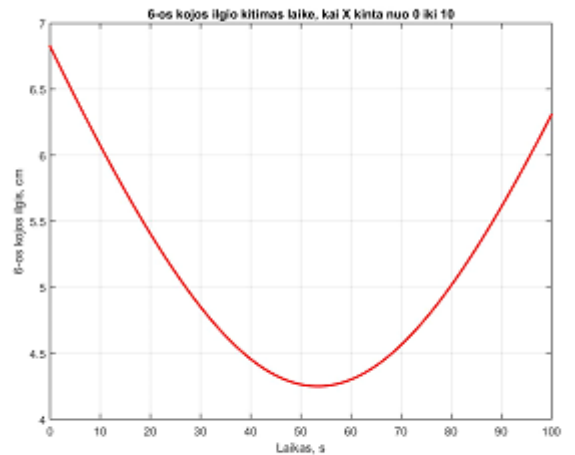


b)

33 pav. Kojų ilgių priklausomybė kintant X ašis kryptimi per tam tikrą laiką, a - 3 koja, b - 4 koja



a)



b)

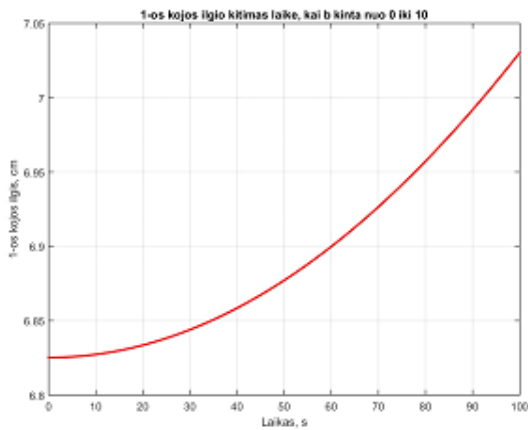
34 pav. Kojų ilgių priklausomybė kintant X ašis kryptimi per tam tikrą laiką, a - 5 koja, b - 6 koja

Judriąją platformą judinant X padėtimi 3-4 kojos stabiliai ilgėja iki maksimalaus X taško 10, 2 ir 5 kojos palaipsniui ilgėja trumpėja, 1-6 kojos iki 5 trumpėja, po to ilgėja iki maksimumo.

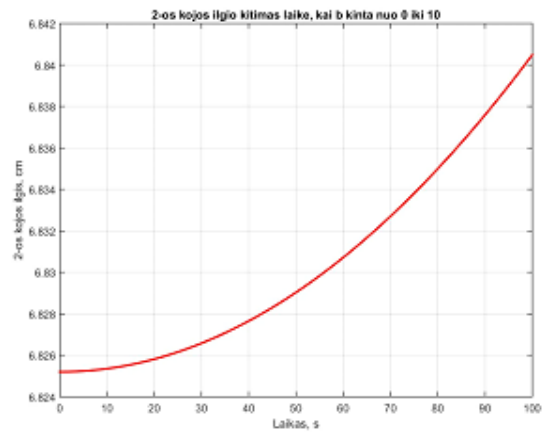
6 lentelė

Kojų ilgio priklausomybė nuo β posūkio kampo

	Kojų ilgiai					
β kampo kitimas	1	2	3	4	5	6
2	6.825	6.825	6.825	6.825	6.825	6.825
6	6.858	6.827	6.879	6.879	6.827	6.858
8	6.899	6.830	6.945	6.845	6.830	6.899
10	6.957	6.835	7.037	7.037	6.835	6.957

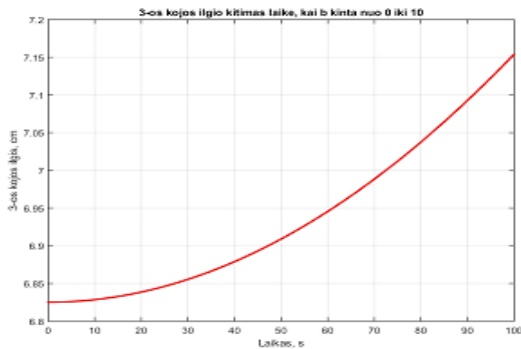


a)

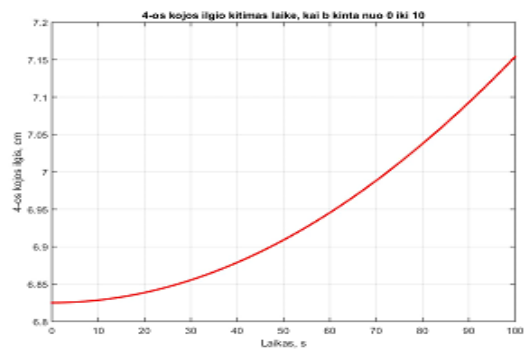


b)

35 pav. Kojų ilgio priklausomybė kintant β posūkio kampui per tam tikrą laiką, a – 1 koja, b – 2 koja



a)



b)

36 pav. Kojų ilgio priklausomybė kintant β posūkio kampui per tam tikrą laiką, a – 3 koja, b – 4 koja

Platformą judinant β posūkio kampu visos kojos stabiliai ilgėja iki maksimalaus 6,5 cm ilgio.

3.5. Kojų jėgų skaičiavimas

Jėgos skaičiuojamos Dwakarath metodu:

$$\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{bmatrix} = inv \begin{bmatrix} l_{1x} & l_{2x} & l_{3x} & l_{4x} & l_{5x} & l_{6x} \\ l_{1y} & l_{2y} & l_{3y} & l_{4y} & l_{5y} & l_{6y} \\ l_{1z} & l_{2z} & l_{3z} & l_{4z} & l_{5z} & l_{6z} \\ (t_1 \times l_1)_x & (t_2 \times l_2)_x & (t_3 \times l_3)_x & (t_4 \times l_4)_x & (t_5 \times l_5)_x & (t_6 \times l_6)_x \\ (t_1 \times l_1)_y & (t_2 \times l_2)_y & (t_3 \times l_3)_y & (t_4 \times l_4)_y & (t_5 \times l_5)_y & (t_6 \times l_6)_y \\ (t_1 \times l_1)_z & (t_2 \times l_2)_z & (t_3 \times l_3)_z & (t_4 \times l_4)_z & (t_5 \times l_5)_z & (t_6 \times l_6)_z \end{bmatrix} \cdot \begin{bmatrix} (F_e)_x \\ (F_e)_y \\ (F_e)_z \\ (M_e)_x \\ (M_e)_y \\ (M_e)_z \end{bmatrix}, \quad (28)$$

Šiam skaičiavimui MatLab programoje sukuriama atskira funkcija. (4 priedas)

Nurodomos norimos statinės vertės. Šiuo atveju nurodoma, jog Z visada yra 10. Tai negalioja, jei Z aprašomas kaip kintamasis.

Jei kitimas vyksta nuo 0 iki 2, kas 0.1 kaip mūsų pavyzdyje, tai masyvas p:

p = 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2
1.3 1.4 1.5 1.6 1.7 1.8 1.9 2

p įdedamas į atitinkamą masyvo p_m eilutę. Šiuo atveju tai kintamasis X, todėl tai pirma eilutė.

7 lentelė

Padėčių ir posūkio kampų masyvas

0	0.	0.	0.	0.	0.	0.	0.	0.	0.	1	1.	1.	1.	1.	1.	1.	1.	1.	1.	2
	1	2	3	4	5	6	7	8	9		1	2	3	4	5	6	7	8	9	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	10	10	10	10	10	10	10	10	10	1	10	10	10	10	10	10	10	10	10	1
0										0										0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

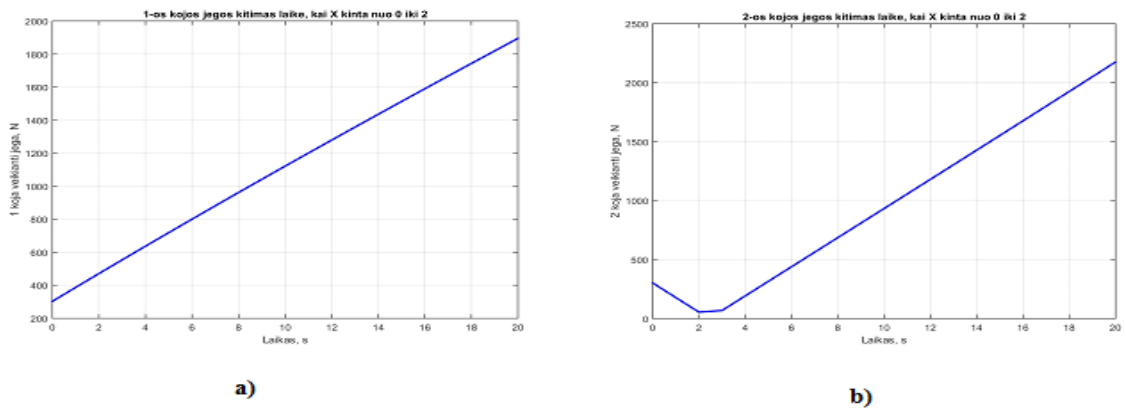
Pačioje tyrimo pradžioje aprašomos išorinės jėgos. Kadangi platformos masė yra 150kg, tai jėga yra 1500N.

Pirmiausia apskaičiuojamos platformos koordinatės, po to jos duodamos funkcijai „gauti_koju_jegas”, kartu su išorinių jėgų kintamaisiais.

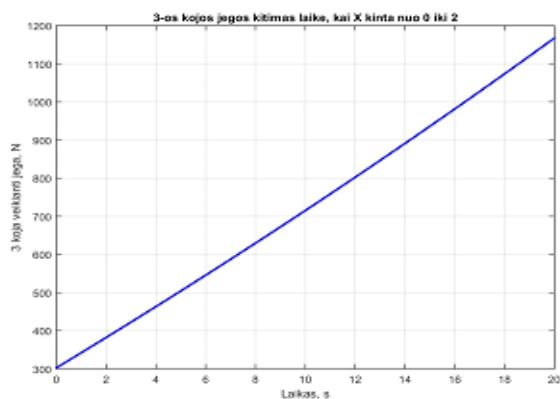
Gauti duomenys saugojami Fk masyve:

Fk <21x6 double>						
	1	2	3	4	5	6
1	302.6790	302.6790	302.6790	302.6790	302.6790	302.6790
2	387.0617	178.6469	342.2916	342.2916	178.6469	387.0617
3	470.8503	54.7810	382.2550	382.2550	54.7810	470.8503
4	554.0595	68.9440	422.5745	422.5745	68.9440	554.0595
5	636.7042	192.5534	463.2552	463.2552	192.5534	636.7042
6	718.7994	316.0726	504.3024	504.3024	316.0726	718.7994
7	800.3603	439.5269	545.7211	545.7211	439.5269	800.3603
8	881.4024	562.9417	587.5161	587.5161	562.9417	881.4024
9	961.9413	686.3425	629.6924	629.6924	686.3425	961.9413
10	1.0420e+03	809.7548	672.2547	672.2547	809.7548	1.0420e+03
11	1.1216e+03	933.2039	715.2077	715.2077	933.2039	1.1216e+03
12	1.2007e+03	1.0567e+03	758.5559	758.5559	1.0567e+03	1.2007e+03
13	1.2794e+03	1.1803e+03	802.3039	802.3039	1.1803e+03	1.2794e+03
14	1.3576e+03	1.3040e+03	846.4561	846.4561	1.3040e+03	1.3576e+03
15	1.4355e+03	1.4279e+03	891.0168	891.0168	1.4279e+03	1.4355e+03
16	1.5130e+03	1.5519e+03	935.9902	935.9902	1.5519e+03	1.5130e+03
17	1.5901e+03	1.6761e+03	981.3805	981.3805	1.6761e+03	1.5901e+03
18	1.6668e+03	1.8005e+03	1.0272e+03	1.0272e+03	1.8005e+03	1.6668e+03
19	1.7432e+03	1.9252e+03	1.0734e+03	1.0734e+03	1.9252e+03	1.7432e+03
20	1.8193e+03	2.0501e+03	1.1201e+03	1.1201e+03	2.0501e+03	1.8193e+03
21	1.8951e+03	2.1753e+03	1.1672e+03	1.1672e+03	2.1753e+03	1.8951e+03

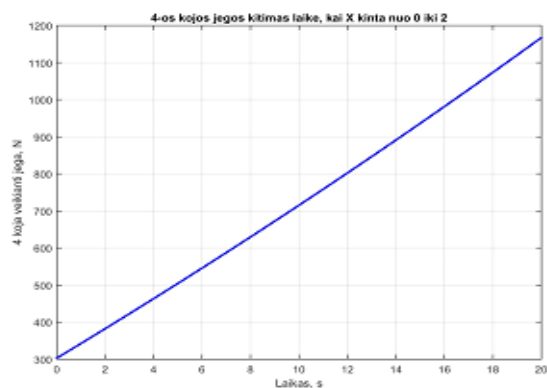
37 pav. Fk masyvas, kuris sukurtas MatLab programoje



38 pav. Kojas veikiančių jėgų kitimas laike keičiantis X padėčiai. a – 1 koja, b – 2 koja

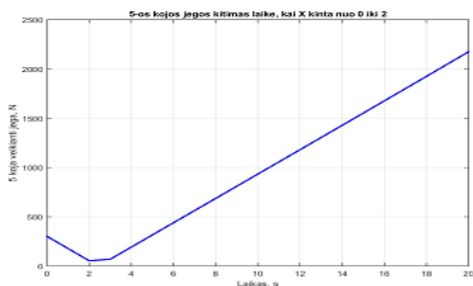


a)

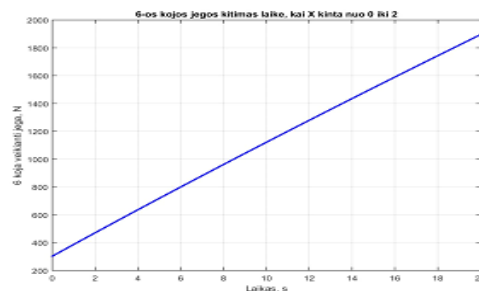


b)

39 pav. Kojas veikiančių jėgų kitimas laike keičiantis X padėčiai. a – 3 koja, b – 4 koja



a)



b)

40 pav. Kojas veikiančių jėgų kitimas laike keičiantis X padėčiai. a – 5 koja, b – 6 koja

Šiuo atveju mažiausia jėga 5 ir 2 manipulatoriaus kojų būna 2-ą judėjimo sekundę, po to jėga tolygiai didėja visų kojų vienodai.

3.6 Skyriaus išvados

1. Parinkta geometrinė konstrukcija Stiuarto platformai, kurios duomenys bvo toliau naudojami sudarant , sprendžiant lygtis ir toliau modeliuojant MatLab programos pakete.

2. Tam, kad būtų lengviau dirbti ir orientuotis, programa buvo padalinta į atskirus failus. Programa susideda iš 5 matlab kodo failų, grafinio vartotojo sąsajos lango ir 3 duomenų failų.

3. Jau anksčiau aptartomis formulėmis pasinaudojus MatLab programos pakete sukurtas Stiuarto platformos geometrinis modelis, kuris vėliau buvo analizuojamas.

4. Analizuojant Stiuarto platformą išsiaiškinta, jog kojų ilgiai labiausiai priklausomi nuo padėties kitimų X,Y,Z koordinačių sistemose, tuo tarpu keičiantis posūkio kampams tokio didelio skirtumo nestebėta.

5. Apskaičiavus Stiuarto platformos kojas veikiančias jėgas, paaiškėjo jog didžiausia veikianti jėga siekia 1500N, mažiausia vos 300 N.

IŠVADOS

1. Atlikus literatūros analizę, nustatyta, kad Stiuarto platforma kaip ir daugelis lygiagrečios kinematinės konstrukcijos robotų, nesudėtingu atvirkštiniu kinematinį padėčių uždaviniu. Kuris leidžia apskaičiuoti reikiamus vykdyklių ilgius ir taip valdyti mechanizmą.
2. SimulinkMechanic programos pakete tirtas Stiuarto platformos valdymo modelis, sukurtas matematinis/mechaninis modelis – tačiau tyrimo eigoje susidurta su kliūtimis, kurios neleidavo tiksliai apskaičiuoti norimos trajektorijos, jėgų, sąnarų apkrovų – programa strikdavo. Judesys būdavo, tačiau laikinai – todėl skaičiavimai laikyti netikslūs ir nepateikiami į baigiamojo darbo išvadas.
3. Didžiausias dėmesys buvo skirtas Matlab programos kūrimui, sukurtas erdvinis modelis, kurio apskaičiuotos ribinės reikšmės X,Y,Z koordinačių sistemoje, minimalios ir maksimalios kojų ilgių reikšmės, judėjimo trajektorijos, jėgos.
4. Pateiktos padėties ir judesio matricos, kurios reikalingos Stiuarto platformos judėjimui, judesio programavimui bei naudojant Matlab programą apskaičiuotos erdvės ribos, koordinačių Sistema, kuriose gautas modelis gali nevaržomai judėti, apskaičiuotos jėgos veikiančios platformos kojas – analizuojant paaiškėjo, jog mažiausia jėga veikia 5- 2 kojas nepriklausomai nuo kintamos padėties koordinačių sistemoje.

LITERATŪROS SĄRAŠAS

1. SANDRIA REBORT „Some Approaches for Modeling and Analysis of a Parallel Mechanism with Stewart Platform Architecture”;
2. MAGNUS BERTHELTSEN KJELLAND Design of a Heave Compensation System with a Redundant Hydraulic Manipulator ; 2011;
3. LEONARDO JELENCOVIC Kinematic evaluation and forward kinematic problem for Stewart platform based manipulators;
4. M. UCHIYAMA , ET AL. "Performance Evaluation of manipulators using the Jacobian and its applications to trajectory planning";
5. Merlet J.-P. Parallel Robots (Second Edition). Springer, SophiaAntipolis, France 2006, p. 8-13.
6. MADSEN L. A., KRISTENSEN G. S. Design of Stewart Platform for Wave Compensation. Masters Thesis. Aalborg Universitat, 2012. Prieiga per internetą: < http://projekter.aau.dk/projekter/files/63502229/EMSD415a_Final.pdf>. [žiūrėta 2016 vasario 12 d.].
7. RoboCrane. Prieiga per internetą: < <http://www.cs.cmu.edu/~Xavier/robocrane.html>> [žiūrėta 2017 vasario 12 d.].
8. KÖVÁRI A. Real-Time Modeling of an Electro-hydraulic Servo System // Computational Intelligence in Engineering, SCI 313. Springer-Verlag, Berlin, Heidelberg, 2010, p. 301-311.
9. SOHL G. A., BOBROW E. J. Experiments and Simulations on the Nonlinear Control of a Hydraulic Servosystem // Control Systems Technology, IEEE Transactions on (Volume: 7, Issue: 2). Dept. of Mech. & Aerosp. Eng., California Univ., Irvine, CA, USA, 1999, p. 178-187.
10. RYDBERG K. E. Hydraulic servo systems. Linköpings universitet, IEI // Fluid and Mechanical Systems. Prieiga per internetą: < https://www.iei.liu.se/flumes/tmhp51/filearchive/coursematerial/1.105708/HydServoSystems_part1.pdf > [žiūrėta 2017-03-14].
11. LEVIŠAUSKAS D. Automatinio reguliavimo sistemų derinimas. Vilniaus pedagoginio universiteto leidykla, Vilnius, 2008, p. 119. e-ISBN 978-609-02-0376-7.
12. JIN Y., CHANAL H., PACCOT F. Parallel Robots. Handbook of Manufacturing Engineering and Technology. Springer-Verlag London, 2015, p. 2091-2168. e ISBN 978-1-4471-4670-4
13. Stewart platform. Prieiga per internetą: < http://en.wikipedia.org/wiki/Stewart_platform > [žiūrėta 2016-10-27].
14. Julien B., Ilian B., Bruno M., Se’bastien B. Jason M. L., Claude P. Kinematic characterisation of hexapods for industry // Industrial Robot: An International Journal 37/1. Emerald Group Publishing Limited, 2010, p. 79– 88.
15. Stewart platform. Prieiga per internetą: < https://en.wikipedia.org/wiki/Stewart_platform> [žiūrėta 2016-04-01].
16. Stiuarto platforma. Prieiga per internetą < www.parallemic.org> [Žiūrėta : 2016-10-25]
17. Stiuarto platforma. Prieiga per internetą <www.mdpi.com> [Žiūrėta : 2016-11-25]
18. Stiuarto platforma. Prieiga per internetą < robotics.technion.ac.il> [Žiūrėta : 2016-11-25]
19. Stiuarto platforma. Prieiga per internetą < www.iaeng.org> [Žiūrėta : 2016-11-25]

20. Stiuarto platforma. Prieiga per internetą < www.zyvex.com > [Žiūrėta : 2016-10-25]

PRIEDAI

1. PRADINĖS REIKŠMĖS

```

clear all % Išvalomas darbalaukis

Bc = 11; % Bazes parametras "c" pagal uzduoti
Bd = 22; % Bazes parametras "d" pagal uzduoti

Pc = 6; % Platformos parametras "c" pagal uzduoti
Pd = 13.5; % Platformos parametras "d" pagal uzduoti

h = 0; % Pradinis platformos aukstis

sq3 = sqrt(3); % Skaiciaus 3 kvadratine saknis

% Bazes tasku koordinates
% X vertes % Y vertes % Z vertes
B(1, 1) = sq3*((2*Bc)+Bd)/6; B(2, 1) = Bd/2; B(3, 1) = 0;
B(1, 2) = -sq3*(Bc-Bd)/6; B(2, 2) = (Bc+Bd)/2; B(3, 2) = 0;
B(1, 3) = -(sq3*(Bc+(2*Bd)))/6; B(2, 3) = Bc/2; B(3, 3) = 0;
B(1, 4) = -(sq3*(Bc+(2*Bd)))/6; B(2, 4) = -Bc/2; B(3, 4) = 0;
B(1, 5) = -sq3*(Bc-Bd)/6; B(2, 5) = -(Bc+Bd)/2; B(3, 5) = 0;
B(1, 6) = sq3*((2*Bc)+Bd)/6; B(2, 6) = -Bd/2; B(3, 6) = 0;

% Platformos tasku koordinates
% X vertes % Y vertes % Z vertes
P(1, 1) = sq3*((2*Pc)+Pd)/6; P(2, 1) = Pd/2; P(3, 1) = 0;
P(1, 2) = -(sq3*(Pc-Pd))/6; P(2, 2) = (Pc+Pd)/2; P(3, 2) = 0;
P(1, 3) = -(sq3*(Pc+(2*Pd)))/6; P(2, 3) = Pc/2; P(3, 3) = 0;
P(1, 4) = -(sq3*(Pc+(2*Pd)))/6; P(2, 4) = -Pc/2; P(3, 4) = 0;
P(1, 5) = -sq3*(Pc-Pd)/6; P(2, 5) = -(Pc+Pd)/2; P(3, 5) = 0;
P(1, 6) = sq3*((2*Pc)+Pd)/6; P(2, 6) = -Pd/2; P(3, 6) = 0;

% Issaugoma platformos geometrija
save('Platformos geometrija', 'P', 'B')

% Nustatomos pradines pozicijos ir posukio kampo reiksmes:
X0 = 0;
Y0 = 0;
Z0 = 0;

a0 = 0;
b0 = 0;
y0 = 0;

% Issaugoma dabartine pozicija ir posukio kampai
save('Dabartine pozicija', 'X0', 'Y0', 'Z0', 'a0', 'b0', 'y0');

% Pirmines ribos pozicijos ir posukio nustatymui
Xmin = -10;
Xmax = 10;
Ymin = -10;
Ymax = 10;
Zmin = 0;

```

```

Zmax = 20;

amin = -45;
amax = 45;
bmin = -45;
bmax = 45;
ymin = -45;
ymax = 45;

% Ribos issaugomos "Ribos"
save('Ribos', 'Xmin', 'Xmax', 'Ymin', 'Ymax', 'Zmin', 'Zmax', 'amin', 'amax', 'bmin', 'bmax', 'ymin', 'ymax');

% Koju spalvos
% RGB koduoteje (Raudona, Zalia, Melyna)
% spalvos:      R      G      B
KojuSpalvos = [0.3  0.3  0.3; % Pirmos kojos spalva
               0.0  0.5  1.0; % Antros kojos spalva
               0.8  0.8  0.0; % Trecios kojos spalva
               0.0  1.0  1.0; % Ketvirtos kojos spalva
               0.3  0.7  0.4; % Penktos kojos spalva
               0.5  0.0  0.5]; % Sestos kojos spalva
save('KojuSpalvos', 'KojuSpalvos');

t = 60; % grafiku ilgio nustatymas (kiek padeties tasku bus matyti grafike)

Lt(:,t) = [0;0;0;0;0;0]; % Sukuriama koju ilgiu matrica ir uzpildoma nuliais

% Padeties matricos atvaizdavimui grafike:
Xt(:,t) = 0;
Yt(:,t) = 0;
Zt(:,t) = 0;

at(:,t) = 0;
bt(:,t) = 0;
yt(:,t) = 0;

save('koju_ilgiu_matrica', 'Lt');
save('padeties_masyvai', 'Xt', 'Yt', 'Zt');
save('posukio_masyvai', 'at', 'bt', 'yt');

Xprog(1:20) = 0
Yprog(1:20) = 0
Zprog(1:20) = 10
aprogram(1:20) = 0
bprog(1:20) = 0
yprog(1:20) = 0

```


2. GAUTOS PRADINĖS KOORDINATĖS IR KOJŲ ILGIAI

```

function [B,P,L] = gauti_koordinates_ir_koju_ilgius(X,Y,Z,a,b,y)

a = a * pi / 180;    % kampas keičiamas i radianus
b = b * pi / 180;    % kampas keičiamas i radianus
y = y * pi / 180;    % kampas keičiamas i radianus

% Posukio matrica R:
R = [cos(b)*cos(y)+sin(a)*sin(b)*sin(y)  sin(a)*sin(b)*cos(y)- cos(b)*sin(y)
     cos(a)*sin(b);
     cos(a)*sin(y)                        cos(a)*cos(y)
     sin(a);
     sin(a)*cos(b)*sin(y)-sin(b)*cos(y)  sin(b)*sin(y)+sin(a)*cos(b)*cos(y)
     cos(a)*cos(b)];

% Pad?ties matrica T:
T = [X;
     Y;
     Z];

load('Platformos geometrija') % Ikeliamos platformos koordinates

for i = 1:6
    P(:,i) = R * P(:,i) + T; % Apskaiciuojamos platformos koordinates

    Lm(:,i) = P(:,i) - B(:,i); % Apskaiciuojama
koju x,y,z ilgiu matrica
    L(i)=sqrt(Lm(1,i)^2 + (Lm(2,i))^2 + (Lm(3,i))^2); % Apskaiciuojami
koju vektoriu ilgiai
end

end

```

3. GAUTOS TARPINĖS KOJŲ PADĖTYS

```

function [X,Y,Z,a,b,y] = gauti_tarpines_padetis(X1,Y1,Z1,a1,b1,y1,s)

load('Dabartine pozicija');

difX = X1 - X0;           % skirtumas tarp buvusios ir nustatomos X padeties
if difX == 0              % jei skirtumo nera
    X(1:s) = X1;          % seka uzpildoma dabartine reiksme
else                      % jei skirtumas yra
    incX = difX/(s-1);    % X pozicijos kitimo dydis
    X = X0:incX:X1;      % X pozicijos kitimo sekos matrica
end

difY = Y1 - Y0;           % skirtumas tarp buvusios ir nustatomos Y padeties
if difY == 0              % jei skirtumo nera
    Y(1:s) = Y1;          % seka uzpildoma dabartine reiksme
else                      % jei skirtumas yra
    incY = difY/(s-1);    % Y pozicijos kitimo dydis
    Y = Y0:incY:Y1;      % Y pozicijos kitimo sekos matrica
end

difZ = Z1 - Z0;           % skirtumas tarp buvusios ir nustatomos Z padeties
if difZ == 0              % jei skirtumo nera
    Z(1:s) = Z1;          % seka uzpildoma dabartine reiksme
else                      % jei skirtumas yra
    incZ = difZ/(s-1);    % Z pozicijos kitimo dydis
    Z = Z0:incZ:Z1;      % Z pozicijos kitimo sekos matrica
end

difa = a1 - a0;           % skirtumas tarp buvusio ir nustatomo a posukio
kampe
if difa == 0              % jei skirtumo nera
    a(1:s) = a1;          % seka uzpildoma dabartine reiksme
else                      % jei skirtumas yra
    inca = difa/(s-1);    % a kampe kitimo dydis
    a = a0:inca:a1;      % a kampe kitimo sekos matrica
end

difb = b1 - b0;           % skirtumas tarp buvusio ir nustatomo b posukio
kampe
if difb == 0              % jei skirtumo nera
    b(1:s) = b1;          % seka uzpildoma dabartine reiksme
else                      % jei skirtumas yra
    incb = difb/(s-1);    % b kampe kitimo dydis
    b = b0:incb:b1;      % b kampe kitimo sekos matrica
end

difgamma = gamma1 - gamma0; % skirtumas tarp buvusio ir nustatomo gamma posukio
kampe
if difgamma == 0          % jei skirtumo nera
    gamma(1:s) = gamma1; % seka uzpildoma dabartine reiksme
else                      % jei skirtumas yra
    incgamma = difgamma/(s-1); % gamma kampe kitimo dydis
    gamma = gamma0:incgamma:gamma1; % gamma kampe kitimo sekos matrica
end

```

```
        incy = dify/(s-1); % y kampo kitimo dydis
        y = y0:incy:y1;    % y kampo kitimo sekos matrica
end

X0 = X(s);
Y0 = Y(s);
Z0 = Z(s);

a0 = a(s);
b0 = b(s);
y0 = y(s);

save('Dabartine pozicija','X0','Y0','Z0','a0','b0','y0');

end
```

4. KOJŲ JĖGŲ SKAIČIAVIMAS

Fk masyvo duomenys atvaizduojami grafikuose

```

for i=1:t % ciklas pagal elementu skaiciu ( i kinta nuo 1 iki s)
    [B, P, L(i,:)] = gauti_koordinates_ir_koju_ilgius(pm(X,i), pm(Y,i), pm(Z,i), pm(a,i), pm(b,i), pm(y,i));
    Fk(i,:) = gauti_koju_jegas(pm(X,i), pm(Y,i), pm(Z,i), B, P, F, M);
end

asis = Z; % priklausomybe nuo asies Z (pakeisti i norima)
pp = ['X','Y','Z','a','b','y']; % padeciu pavadinimai
s = 0:(t-1);

for nr=1:6
    figure(nr)
    plot(s,Fk(:,nr), '-b', 'LineWidth',2) % L(:,1) dvitaskis ir skaicius 1 reiskia, kad mus domina tik stulpelis
nr 1
    title(sprintf('%d-os kojos jegos kitimas laike, kai %c kinta nuo %d iki %d', nr, pp(k), p0, p1)) % grafiko
pavadinimas
    xlabel('Laikas, s') % x asies pavadinimas
    ylabel(sprintf('%d koja veikianti jega, N', nr)) % y asies pavadinimas
    grid on % ijungiamas tinklelis
    saveas(gcf,sprintf('%d-os kojos jegos kitimas laike, kai %c kinta nuo %d iki %d.jpg', nr, pp(k), p0, p1))
end

% visu koju atvaizdavimas
figure(nr+1)
hold on
plot(s,Fk(:,1), '-ro') % L(:,1) dvitaskis ir skaicius 1 reiskia, kad mus domina tik stulpelis nr 1
plot(s,Fk(:,2), '-bx')
plot(s,Fk(:,3), '-k+')
plot(s,Fk(:,4), '-gs')
plot(s,Fk(:,5), '-cd')
plot(s,Fk(:,6), '-mh')
hold off

title(sprintf('Koju jegu priklausomybe, kai %c kinta nuo %d iki %d', pp(k), p0, p1)) % grafiko pavadinimas
xlabel('Laikas, s') % x asies pavadinimas
ylabel('Kojas veikiancios jegos, N') % y asies pavadinimas
grid on % ijungiamas tinklelis
legend('Koja 1', 'Koja 2', 'Koja 3', 'Koja 4', 'Koja 5', 'Koja 6')
legend('Location','BestOutside') % legenda
saveas(gcf,sprintf('Koju jegu priklausomybe, kai %c kinta nuo %d iki %d.jpg', pp(k), p0, p1))

```