



Kauno technologijos universitetas

Elektros ir elektronikos fakultetas

Straublio tipo roboto valdymo algoritmo sukūrimas ir tyrimas

Baigiamasis magistro projektas

Mindaugas Matukaitis

Projekto autorius

Doc. dr. Renaldas Urniežius

Vadovas

Kaunas, 2020



Kauno technologijos universitetas

Elektros ir elektronikos fakultetas

Straublio tipo roboto valdymo algoritmo sukūrimas ir tyrimas

Baigiamasis magistro projektas

Valdymo technologijos (6211EX014)

Mindaugas Matukaitis

Projekto autorius

Doc. dr. Renaldas Urniežius

Vadovas

Doc. dr. Virginijus Baranauskas

Recenzentas

Kaunas, 2020



Kauno technologijos universitetas

Elektros ir elektronikos fakultetas

Mindaugas Matukaitis

Straublio tipo roboto valdymo algoritmo sukūrimas ir tyrimas

Akademinio sąžiningumo deklaracija

Patvirtinu, kad mano, Mindaugo Matukaičio, baigiamasis projektas tema „Straublio tipo roboto valdymo algoritmo sukūrimas ir 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)

Matukaitis, Mindaugas. Straublio tipo roboto valdymo algoritmo sukūrimas ir tyrimas. Magistro baigiamasis projektas / vadovas doc. dr. Renaldas Urniežius; Kauno technologijos universitetas, Elektros ir elektronikos fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Elektronikos inžinerija (inžinerijos mokslai).

Reikšminiai žodžiai: straublio tipo robotas, valdymas, programavimas.

Kaunas, 2020. 62 p.

Santrauka

Magistro baigiamojo darbo tikslas – sukurti straublio tipo roboto trijų segmentų valdymo algoritmą, kuris, nurodžius užduoties taško koordinates ir trečiojo segmento galo krypties vektorių, roboto darbo zonoje surastų visų trijų segmentų išsilankstymą ir pasukimo kampus.

Magistro darbe yra išanalizuojama straublio tipo roboto konstrukcija, konstrukciniai variantai ir veikimo principas. Apžvelgiami straublio tipo roboto tiesioginės ir atvirkštinės kinematikos uždaviniai ir jų sprendimo metodai. Išsprendžiamas straublio tipo roboto trijų segmentų atvirkštinės kinematikos uždavinys dvimatėje ir trimatėje koordinačių sistemose, pritaikant vektorių savybes. Pagal išspręstą atvirkštinį kinematikos uždavinį, pritaikant vektorių savybes, sukuriamas straublio tipo roboto, turinčio 3 segmentus, valdymo algoritmas ir ištiriama jo greಿತaveika bei tikslumas.

Matukaitis, Mindaugas. Development and Research of a Trunk-type Robot Control Algorithm. Master's Final project / supervisor doc. dr. Renaldas Urniežius; Kaunas University of Technology, Faculty of Electrical and Electronics Engineering.

Study field and area (study field group): Electronics Engineering (Engineering Science).

Keywords: trunk-type robot, control, programming.

Kaunas, 2020. 62 p.

Summary

The aim of the master's final project is to present the control algorithm of the three segments trunk type robot, which, by specifying the coordinates of the task point and the third segment orientation, would find the bending and rotation angles of all three robot segments.

This final project analyzes the construction, different methodology approaches and working principle of a trunk type robot. This text presents forward and inverse kinematics and related methods of the trunk type robot. Author proposes a vector-based solution to solve three-segment trunk type robot inverse kinematics problem in a two-dimensional and three-dimensional coordinates systems. The three-segment trunk type robot control algorithm uses vector method approach for solving the inverse kinematics problem.

Turinys

| | |
|---|-----------|
| Įvadas | 7 |
| 1. Literatūros apžvalga | 8 |
| 1.1. Straublio tipo roboto tipai, konstrukcija ir veikimo principas..... | 8 |
| 1.2. Straublio tipo roboto valdymo vykdymo įtaisų rūšys..... | 10 |
| 1.2.1. Bešepetėlinio nuolatinės srovės elektros variklio struktūra, jos tipai, veikimo principas ir valdymo metodai | 10 |
| 1.3. „Sausgyslėmis“ valdomo straublio tipo roboto segmentų konstrukciniai variantai ir „sausgyslių“ skaičius bei išdėstymas segmente | 14 |
| 1.4. Straublio tipo roboto tiesioginės ir atvirkštinės kinematikos uždaviniai..... | 19 |
| 1.4.1. Pagrindiniai straublio tipo roboto segmento parametrai ir Oilerio kampai | 19 |
| 1.4.2. Straublio tipo roboto tiesioginės kinematikos uždavinys ir jo sprendimo metodai..... | 21 |
| 1.4.3. Straublio tipo roboto atvirkštinės kinematikos uždavinys ir jo sprendimo metodai | 23 |
| 2. Straublio tipo roboto valdymo algoritmo kūrimo metodika | 26 |
| 2.1. Straublio tipo roboto valdymo algoritmą apibrėžiančios sąlygos | 26 |
| 2.2. Straublio tipo roboto segmentų „sausgyslių“ ilgių skaičiavimo formulė ir valdymo algoritmo ieškomų parametrų apibrėžimas..... | 26 |
| 2.3. Straublio tipo roboto valdymo algoritmo kūrimo pradžia pritaikant vektorius..... | 28 |
| 2.4. Straublio tipo roboto 3-jų segmentų išsilenkimo kampų paieškos uždavinys dvimatėje plokštumoje | 29 |
| 2.5. Straublio tipo roboto 3-jų segmentų išsilenkimo ir pasukimo kampų paieškos uždavinys trimatėje aplinkoje ir roboto valdymo algoritmo kūrimas..... | 39 |
| 3. Straublio tipo roboto valdymo algoritmo tyrimo rezultatai | 45 |
| 3.1. Valdymo algoritmo greitaveikos tyrimo rezultatai..... | 45 |
| 3.2. Roboto išsilankstymo trimatėje erdvėje, pagal valdymo algoritmo surastų roboto segmentų pasukimo ir sulenkimo kampų dydžius, pozicionavimo tikslumo tyrimo rezultatai..... | 51 |
| Rezultatai ir išvados | 60 |
| Literatūros sąrašas | 61 |

Įvadas

Šiuolaikinėje pramonėje automatizuojant įvairius gamybos procesus vis dažniau yra taikomi įvairių tipų robotai, taip pakeičiant žmogaus darbo jėgą ir panaikinant rutininius, fiziškai ir protiškai varginančius darbus. Priklausomai nuo proceso, taikomi įvairių tipų robotai, pradedant nuo stacionarių, darbo vietos nekeičiančių robotų ir baigiant mobiliais robotais, gebančiais judėti erdvėje. Vis dėlto, dažniausiai pramonės gamybos procesuose darbai, kur žmones pakeičia robotai – manipulatoriai yra stacionariose vietose ir šiems darbams atlikti dažniausiai taikomi stacionarūs, n ašių turintys robotai manipulatoriai. Vienas iš pavyzdžių būtų metalo apdirbimo, automobilių pramonėje sutinkami uždaviniai, kur dažnai yra taikomi robotai manipulatoriai, tai yra dažymo arba detalės paviršiaus padengimo uždaviniai. Tačiau ar galima pritaikyti šiai užduočiai įvykdyti straublio tipo robotą?

Straublio tipo robotai dažniausiai yra sutinkami šiek tiek kitokio tipo srityse, negu n ašių turintys robotai – manipulatoriai, tai yra žmonių gelbėjimo operacijose ir medicinoje (chirurginėse operacijose). Straublio tipo robotai nuo n ašių turinčių robotų manipuliatorių skiriasi savo lankstumu ir lengva konstrukcija, kadangi visi vykdykliai, valdantys straublio tipo robotą, yra sumontuojami roboto bazėje, o ne ant pačio roboto, kaip kad robotuose manipuliatoriuose. Be to, dažymo uždavinyje roboto keliamoji galia neprivalo būti didelė, svarbiausia, jog robotas sugebėtų orientuoti dažymo įrankio purkštuką erdvėje. Todėl šis magistro darbas yra skirtas sukurti straublio tipo robotui skirtą valdymo algoritmą, pagal kurį trijų segmentų roboto galinis segmentas būtų orientuojamas trimatėje erdvėje ir sugebėtų pasiekti iš anksto užduoto taško koordinates.

Darbo tikslas – sukurti straublio tipo roboto trijų segmentų valdymo algoritmą, kuris, nurodžius užduoties taško koordinates ir trečiojo segmento galo krypties vektorių, roboto darbo zonoje surastų visų trijų segmentų išsilankstymą ir pasukimo kampus.

Darbo uždaviniai:

1. išanalizuoti straublio tipo roboto konstrukciją, konstrukcinius variantus, veikimo principą;
2. apžvelgti straublio tipo roboto tiesioginės ir atvirkštinės kinematikos uždavinius ir jų sprendimo metodus;
3. išspręsti straublio tipo roboto trijų segmentų atvirkštinės kinematikos uždavinį dvimatėje ir trimatėje koordinačių sistemose, pritaikant vektorių savybes;
4. ištirti sukurto valdymo algoritmo greitaveiką ir tikslumą.

Šis magistro darbas yra sudarytas iš trijų pagrindinių skyrių. Pirmajame skyriuje – literatūros apžvalga, kurioje įvykdomi pirmieji du darbo uždaviniai, išvardyti aukščiau. Antrajame skyriuje – metodika, kurioje išsprendžiamas straublio tipo roboto trijų segmentų atvirkštinės kinematikos uždavinys dvimatėje ir trimatėje koordinačių sistemose ir pagal tai sukuriamas straublio tipo roboto valdymo algoritmas. Trečiajame skyriuje – pateikiami atlikti sukurtos straublio tipo roboto valdymo programos greitaveikos ir roboto pozicionavimo tikslumo, pagal valdymo programos apskaičiuotus roboto segmentų sulenkimo ir pasukimų kampus, tyrimų rezultatai. Išvados trumpai apibendrinama, kas pastebėta darant šį baigiamąjį darbą.

1. Literatūros apžvalga

1.1. Straublio tipo roboto tipai, konstrukcija ir veikimo principas

Straublio tipo robotas (angl. *trunk-type robot*) yra lankstus, savo forma primenantis dramblio straublį, robotas, todėl dar jis vadinamas dramblio straublio robotu (angl. *elephant's trunk robot*) arba nepertraukiamo stuburo robotu (angl. *continuum robot*). Straublio tipo robotas savo forma primena gyvatės tipo robotus (angl. *snake type robot*), tačiau „trunk“ ir „snake“ tipo robotai skiriasi savo konstrukcija, dinamika, taip pat paskirtimis. „Snake“ tipo robotas yra mobilus robotas, o „trunk“ robotas yra stacionarus robotas manipulatorius, pritvirtintas prie pagrindo. Atsižvelgiant į tai, jog gyvatės tipo robotas yra mobilus, nes neturi pagrindo ir juda imituodamas gyvatės judesius, jis dažniausiai naudojamas pasiekti ir pamatyti vietas, į kurias žmogus negali fiziškai patekti, nes yra per didelis arba ne toks lankstus. Straublio tipo roboto paskirtis šiek tiek kitokia. Kadangi jis yra stacionarus, jį galima valdyti tiksliau negu „snake“ tipo robotą, todėl uždėjus straublio tipo robotui galinį efektorių, tai yra, įrankį, pavyzdžiui, griebtuvą, galima su juo atlikti tokius darbus, kaip daiktų perkėlimas iš vienos vietos į kitą. Šiuo metu straublio tipo robotai yra dažniausiai naudojami gelbėjimo operacijose ir medicinoje.[6]

Straublio tipo robotai pagal konstrukciją ir veikimo principą yra skirstomi į tris tipus.

1) „Sausgyslėmis“ valdomas robotas (angl. *tendon-driven robot*) – robotą sudaro lankstus stuburas, standūs atraminiai diskai ir „sausgyslės“. Šiuo atveju „sausgyslės“ atstoja metalinės vielos, trosai ar kitokios lanksčios, bet tvirtos medžiagos siūlai, kuriuos traukiant yra valdomas robotas.[1][6]

2) „Elastinių vamzdelių“ robotas (angl. *concentric tube robot*) – šis robotas sudarytas iš elastingų vamzdelių, kurių skersmenys yra skirtingi. Šie vamzdeliai pagal savo skersmenį yra sukišami vienas į kitą (pradedant nuo mažiausio skersmens ir baigiant didžiausiu) ir tokio roboto judesiai yra valdomi traukiant šiuos vamzdelius už vieno jų galo.[2][6]

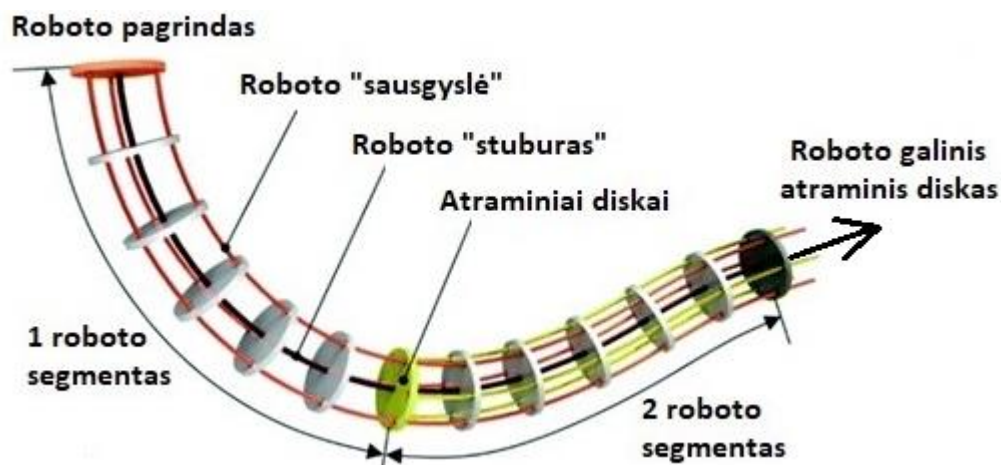
3) Pneumatinis arba hidraulinis nepertraukiamo stuburo robotas (angl. *pneumatically or hydraulically actuated continuum robot*) – tokio tipo robotai yra sudaryti iš elastingų, savo forma gofruotą vamzdį primenančių vamzdžių. Šis robotas valdomas su suslėgtu oru – pneumatiniu būdu, arba su suslėgtu skysčiu – hidraulinio būdu.[3][6]

Šiame darbe tiriamas straublio tipo robotas yra „sausgyslėmis“ valdomo tipo robotas. Šį robotą sudaro 3 esminiai konstrukciniai elementai, kurie yra pavaizduoti 1 paveikslėlyje.

Roboto „stuburas“ – standus, bet elastingas cilindro formos konstrukcinis elementas, kuris išlaiko roboto formą.

Roboto atraminiai diskai – jie skirti metalinėms vieloms arba, kitaip tariant, roboto „sausgyslių“ padėčiai palaikyti palei roboto stuburą.

Roboto „sausgyslės“ – dažniausiai tai būna metalinės vielos, tačiau gali būti ir kitokio tipo medžiagos tvirti, bet lankstūs siūlai arba trosai, kuriuos traukiant arba atleidžiant yra valdomas roboto stuburas.



1 pav. Straublio tipo „sausgyslėmis“ valdomo roboto struktūrinė schema

Iš „sausgyslių“, „stuburo“ ir atraminių diskų yra sudaromas vienas roboto segmentas, kurį galima sulenkti taip, kaip pavaizduota 1-jame paveikslėlyje. Tačiau vieno roboto segmento sulenkimo kampas yra apribotas roboto stuburo standumo. Vieno segmento maksimalus sulenkimo kampas gali kisti labai įvairiai: nuo 45° iki 120° , kampas priklauso nuo segmento konstrukcijos ir ilgio. Taip pat vieno segmento laisvės laipsnių skaičius, priklausomai nuo konstrukcijos, gali būti lygus 2 arba 3 laisvės laipsniams.[4]

Kadangi vienas roboto segmentas turi tik 2 arba 3 laisvės laipsnius, siekiant padidinti roboto laisvės laipsnių skaičių, ant pirmojo roboto segmento galinio atraminio disko galima uždėti kitą segmentą, sudarytą iš tokių pačių elementų (žr. 1 pav.). Dabar 2-ojo segmento galinis atraminis diskas būtų roboto galinis diskas. Tokiu būdu padidinamas ne tik roboto laisvės laipsnių skaičius, bet ir padidėja bendras roboto sulenkimo kampas ir išplečiama roboto darbinė erdvė. Padidinus roboto laisvės laipsnių skaičių bei roboto darbinę erdvę yra praplečiamos roboto pritaikomumo galimybės praktikoje.[5]

Roboto segmentų skaičių galima dar didinti tiek, kiek tai leistų pati roboto konstrukcija. Kol, esant nurodytam segmentų skaičiui, yra išlaikoma kiekviena iš segmento užduota reikiama forma ir padėtis jų sulenkimo metu, robotui nesant apkrautam papildomu svoriu ant galinio atraminio disko, tol nurodytas segmentų skaičius yra leistinas. Norint sukurti robotą iš daugiau negu 3 segmentų yra patartina trumpinti roboto segmentų ilgius, kad būtų lengviau išlaikyti kiekvieno segmento formą, kai atliekamas judesys su robotu.

Didinant segmentų skaičių, sudėtingėja roboto konstrukcija, nes per pirmą roboto segmentą turi praeiti vis daugiau „sausgyslių“, kurios yra skirtos valdyti segmentams, esantiems ant pirmojo segmento. Didėjant roboto „sausgyslių“ skaičiui, didėja pavarų skaičius (1 „sausgyslei“ reikalinga 1 elektros pavara), todėl pridėdant po naują segmentą, didėja roboto savikaina. Tačiau darbo pradžioje apibrėžus roboto darbinę erdvę ir paskirtį, galima parinkti reikiamą roboto segmentų skaičių, parinkti kiekvieno segmento ilgį ir sukonstruoti robotą taip, kad jis neturėtų perteklinių detalių ir mechanizmų jam paskirtoms užduotims atlikti.

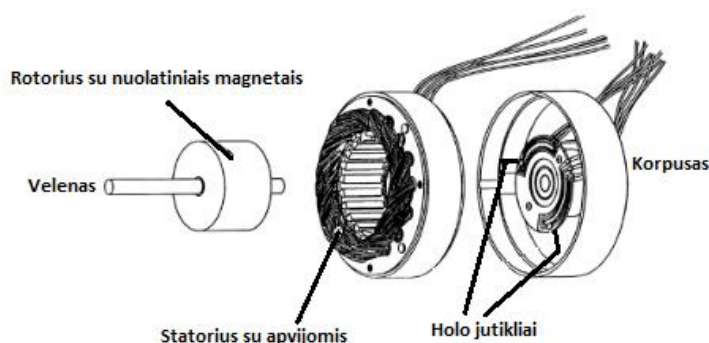
1.2. Straublio tipo roboto valdymo vykdytojų rūšys

Vienas iš esminių roboto konstravimo klausimų yra toks – kokio tipo vykdytojas bus panaudotas konstruojant robotą. Pirmiausiai, vykdytojai skirstomi pagal vartojamos energijos tipą į pneumatinius, hidraulinius ir elektrinius vykdytojus. Kadangi tiriamas robotas yra „sausgyslėmis“ arba, kitaip tariant, metalinėmis vielomis valdomas robotas, konstrukciniu požiūriu elektrinis vykdytojas (elektros variklis) yra geriausias pasirinkimas dėl savo kompaktiškumo, montavimo patogumo, preciziško valdymo, nedidelės kainos.

1.2.1. Bešepetėlinio nuolatinės srovės elektros variklio struktūra, jos tipai, veikimo principas ir valdymo metodai

Dėl bešepetėlinio nuolatinės srovės variklio (angl. *Brushless DC electric motor, BLDC*) privalumų, tokių, kaip didelis sukimo momentas, didelis efektyvumas (nuo 85 % iki 90 %) ir platus greičio reguliavimo diapazonas, šio tipo variklis yra plačiai taikomas įvairiose srityse, pavyzdžiui, pramoninėje automatikoje. Šioje srityje pasitaiko pozicionavimo užduočių. Vienos iš tokių užduočių pavyzdys būtų paprasčiausia 3-jų ašių CNC mašina, kurioje gręžimo įrankis yra judinamas ašių x, y ir z kryptimis (tiesiaiegiai judesiai). CNC mašinų pagrindiniai reikalavimai varikliams: geras stabilumas, platus greičio diapazonas, didelės apkrovos charakteristikos (sistemos apkrovos diapazone, kai apkrova pasikeičia, išėjimo greitis turėtų būti vienodas), aukštas patikimumas. Siekiant geriau suprasti, ar BLDC variklis gali užtikrinti panašius keliamus reikalavimus straublio tipo roboto valdymui, reikia išanalizuoti šio variklio struktūrą, jos tipus, patį variklio veikimo principą, valdymą ir galimus variklio valdymo metodus.[7][8]

Bešepetėlinį nuolatinės srovės variklį sudaro statorius, kuriame yra specialiai suvyniotos apvijos ir rotorius, kuriame yra išdėstyti magnetai (žr. 2 pav.). Priklausomai nuo valdymo metodo, rotoriaus padėties nustatymui gali būti panaudoti holo jutikliai.



2 pav. BLDC variklio struktūra

Statorius yra nejudanti variklio dalis. Statoriuje esančių griovelių skaičius, į kuriuos yra klojamos apvijos, priklauso nuo rotoriaus polių skaičiaus, variklio fazių skaičiaus ir apvijų vyniojimo metodo. Statoriaus apvijos gali būti sujungtos žvaigžde arba trikampiui (kaip ir kintamos srovės asinchroniniai varikliai). Rotorius atlieka variklio sukamąjį judesį, o jame įtvirtintas velenas šį judesį perduoda kitiems mechanizmams. Rotoriuje yra įmontuoti magnetai, kurių skaičius ir dydis daro įtaką variklio sukimo momento jėgai. Tarp rotoriaus ir statoriaus nėra tokių elementų, kaip šepetėliai ar kolektoriai, kurie keltų trintį ir elektrinį kibirkščiavimą. Projektuojant ir konstruojant šį variklį, oro tarpą bei atstumą tarp rotoriaus ir statoriaus stengiamasi sukurti kuo mažesnę, kadangi tai didina variklio

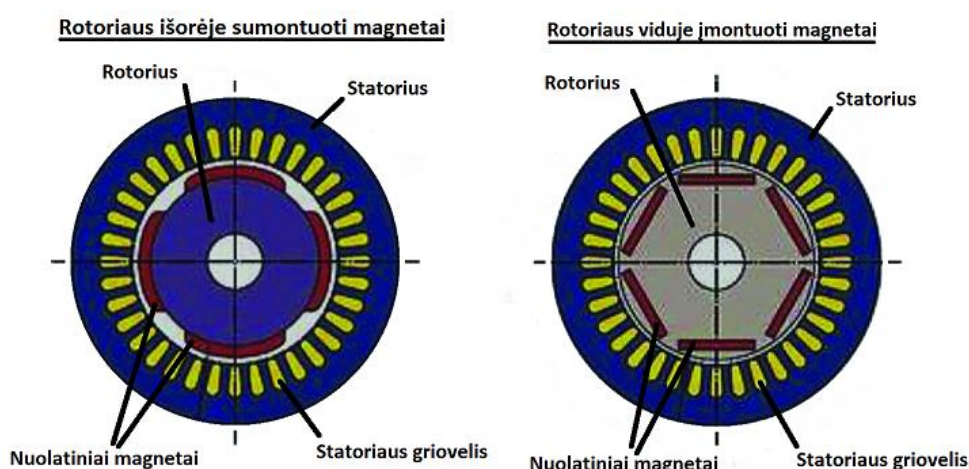
efektyvumą. Standartiškai BLDC variklių oro tarpas siekia nuo 0,3 mm iki 0,8 mm, o labai mažuose varikliuose oro tarpas yra dar mažesnis.[9][10]

BLDC varikliai gali būti skirtingų konstrukcijos tipų ir yra skirstomi pagal:

- konstrukcines medžiagas (nuolatinio magneto tipas);
- nuolatinųjų magnetų montavimą (rotoriaus išorėje arba viduje);
- statoriaus apvijų montavimo metodą (su grioveliais arba be griovelių);
- rotoriaus padėtį variklyje (variklio išorėje arba statoriaus viduje).

Esminiai BLDC variklio konstrukciniai būdai, į kuriuos reikia atkreipti dėmesį renkantis variklį straublio tipo robotui valdyti, yra nuolatinųjų magnetų montavimo vietos variklyje ir rotoriaus galimos padėties variklyje.

Pagrindinis privalumas montuojant nuolatinuos magnetus rotoriaus išorėje (žr. 3 pav.) yra maži priešinės elektrovaros (angl. *back electromotive force, back-EMF*) iškraipymai prie mažos įtampos ir induktyvumo. Tai yra naudinga žemos įtampos varikliams. Tačiau montuojant magnetus rotoriaus išorėje atsiranda tokių problemų, kaip: mažas mechaninis stiprumas, kuris apriboja ir sumažina saugų variklio mechaninį greitį, nėra magnetinės varžos sukimo momento (angl. *magnetic reluctance torque*), taip pat reikia daugiau arba didesnių magnetų montuojant magnetus rotoriaus išorėje, negu montuojant rotoriaus viduje.



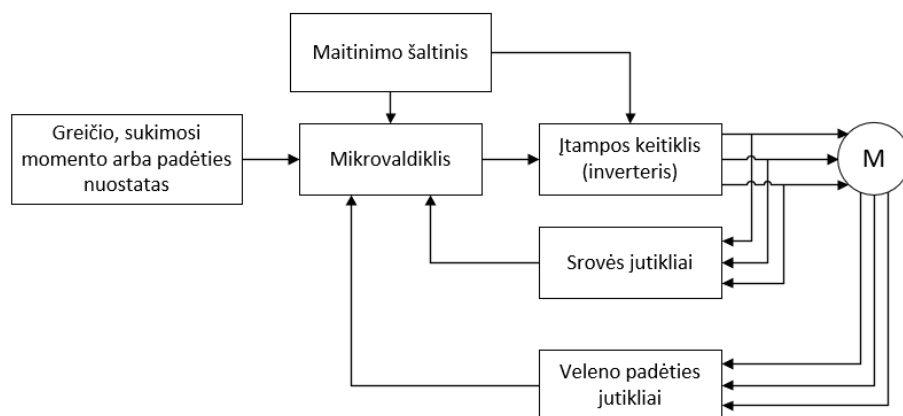
3 pav. BLDC variklio struktūra, kai magnetai yra rotoriaus išorėje arba rotoriaus viduje

BLDC varikliai, kurių rotoriaus viduje (žr. 3 pav.) įmontuoti nuolatiniai magnetai, turi daugiau privalumų negu varikliai, kuriuose magnetai sumontuoti išorėje. Didžiausias privalumas tas, jog įmontavus nuolatinuos magnetus rotoriaus viduje, padidėja rotoriaus mechaninis stiprumas, tvirtumas ir dėl to labai padidėja maksimalus saugus variklio mechaninis greitis. Taip pat yra tiek magnetinės, tiek magnetinės varžos sukimo momentai, todėl, įmontavus į rotoriaus vidų mažiau magnetų, yra gaunama tokia pati variklio išėjimo galia, kaip ir sumontavus daugiau magnetų rotoriaus išorėje.[11]

Bešepetėlinio nuolatinės srovės variklio, kaip ir kitų elektros variklių veikimo principas remiasi ta pačia elektromagnetizmo teorija - kai laidininku tekanti srovė kuria aplink save magnetinį lauką ir šiam sukurtam magnetiniam laukui sąveikaujant su kito laidininko arba magneto sukurtu magnetiniu lauku, jie stumia arba traukia vienas kitą, o kartu su magnetiniais laukais juda ir juos sukūrę

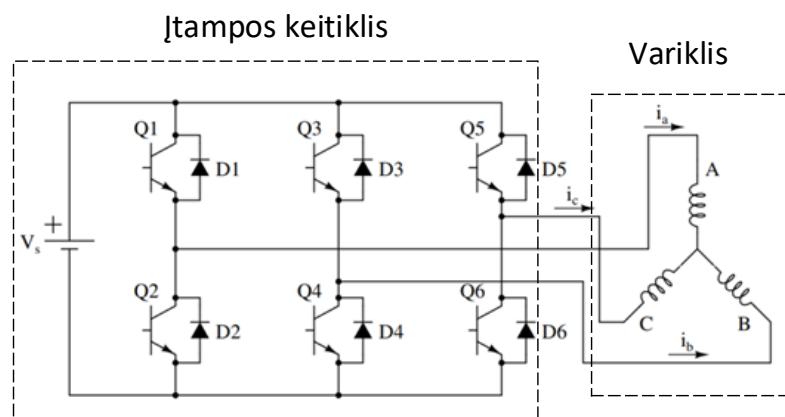
laidininkai arba magnetai. Taip yra atliekamas mechaninis judesys, kuris toliau pritaikius perdavimo mechanizmą yra naudojamas darbui atlikti.[12]

Valdant bešepetėlinius variklius elektroniniu būdu yra reikalinga speciali įranga variklio valdymui. BLDC variklio tipinė valdymo struktūrinė schema yra pavaizduota 4 paveikslėlyje. Šios valdymo schemas pagrindą sudaro: valdomas bešepetėlinis nuolatinės srovės variklis, įtampos keitiklis (inverteris) – paduodantis įtampą į variklį ir mikrovaldiklis, kuris valdo įtampos keitiklį. Mikrovaldikliui informaciją apie variklio padėtį ir paduodamų į apvijas srovės dydžius suteikia sumontuoti jutikliai. Variklio valdymo metodai, algoritmai, parametų nuostatai yra apibrėžiami programuojant mikrovaldiklį.



4 pav. BLDC variklio tipinė valdymo struktūrinė schema

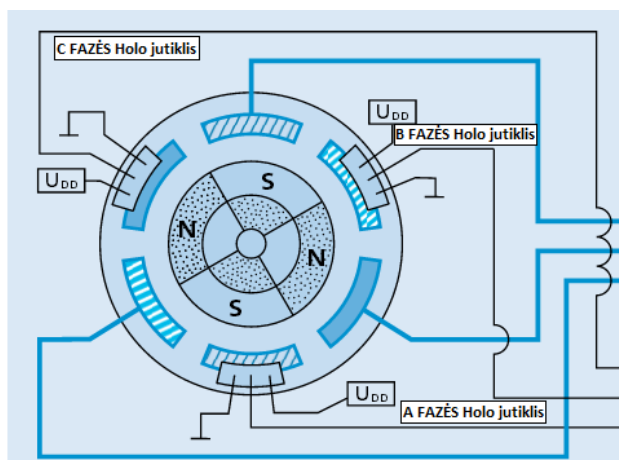
BLDC variklių įtampos valdymas atliekamas valdant įtampos keitiklį, kurį sudaro galios tranzistoriai, pavyzdžiui, lauko tranzistoriai „MOSFET“. Įtampos keitiklio ir variklio schema yra pateikta 5 paveikslėlyje. Tranzistoriai veikia kaip jungikliai, kurie gavę elektrinį signalą praleidžia elektros srovę. Iš anksto apibrėžta seka tranzistorius junginėja mikrovaldiklis.[13]



5 pav. Įtampos keitiklio ir BLDC variklio struktūrinė schema

Tam, kad BLDC variklio velenas sukėtųsi (variklis veiktų) reikia valdyti įtampos keitiklyje esančius tranzistorius, kiekvieną iš jų junginėti atitinkamais laiko momentais, kad apvijose tekančios srovės kurtų tokios krypties magnetinius laukus, kurie sąveikaujant su rotoriuje esančių nuolatinių magnetų sukurtais magnetiniais laukais, sukėtų variklio rotorių. Rotoriuje nuolatinių magnetų kuriami magnetiniai laukai yra pastovių krypčių, todėl, žinant variklio rotoriaus pasisukimo kampą (padėtį), galima tiksliai nurodyti, į kurias variklio apvijas turi būti paduota srovė. Vieni iš labiausiai paplitusių

rotoriaus padėčiai nustatyti skirtų jutiklių yra Holo jutikliai. Dažniausiai padėties jutikliai yra išdėstomi po vieną ant kiekvienos iš BLDC variklio statoriaus maitinimo fazės apvijų vieno poliaus (žr. 6 pav.).[13]



6 pav. Holo jutiklių išdėstymas BLDC variklio statoriuje

Dažniausiai taikomi bešepetėlinių variklių valdymo metodai, kuriems yra reikalinga informacija apie variklio rotoriaus padėtį yra tokie:

Trapecinis valdymas (angl. *Trapezoidal control*) – laikomas pačiu paprasčiausiu bešepetėlinio variklio valdymo metodu. Taip pat šis valdymas pasižymi tuo, jog sukuria didelį variklio sukimosi momentą. Šiuo atveju, nustatčius variklio rotoriaus padėtį su holo jutikliais, įtampos keitiklio tranzistoriai (žr. 5 pav.) yra junginėjami su impulsais. Valdant impulsų signalų plotį, tranzistoriai yra atidaromi tokį laiko tarpą, koks yra impulso plotis. Toks impulso signalo valdymas yra vadinamas impulso pločio moduliacija (angl. *Pulse-Width Modulation, PWM*). Trapecinio valdymo veikimas pagrįstas tuo, jog variklio sukimo momentas yra kuriamas junginėjant tranzistorius taip, kad dvi variklio fazės nuolat sektų viena kitą 60 laipsnių kampų.[13]

Sinusinis valdymas (angl. *Sinusoidal control*) – yra panašus į trapecinį variklio valdymą, nes variklio valdymo signalai yra valdomi pasitelkiant impulso pločio moduliaciją. Tačiau sinusinio BLDC variklio valdymo metodo atveju yra valdomos iškart trys variklio maitinimo fazės, o valdymo signalai yra formuojami sinusinės formos. Keičiant stačiakampių variklio valdymo signalų pločius, yra išgaunamas valdymo signalas, kurio vidutinė srovės vertė atitinka sinusinę signalo formą. Esant tokiam valdymui, yra sumažinami variklio sukimo momento trūkiai, kurie yra būdingi variklį valdant trapeciniu metodu. Tačiau esant sinusiniam valdymui prie didelių greičių, šis metodas pasidaro nebeefektyvus, nes tampa sudėtinga komutuoti valdymo signalą taip, kad jis būtų sinusinės formos.[14]

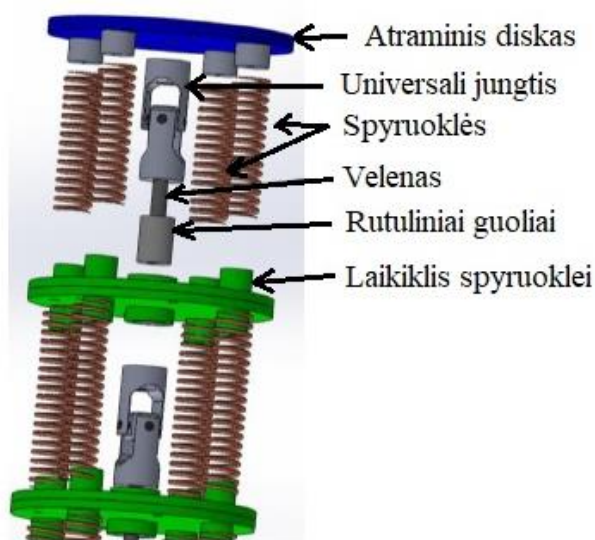
FOC valdymas (angl. *Field Oriented Control, FOC*) – pasitelkiant FOC valdymą varikliui valdyti yra keičiamos trys variklio statoriaus fazių srovės, kurios yra išreikštos erdvės vektoriais. Statoriaus 3-jų fazių srovės yra formulėmis susiejamos su variklio magnetinio lauko srauto ir sukimo momento dalimis. Kiekviena iš trijų fazių srovių yra valdoma atskirai. Variklio valdymo metodas įgyvendinamas atliekant 4 žingsnius: pirmas žingsnis – išmatuoti variklio statoriaus apvijomis tekančias sroves, antras žingsnis – palyginti išmatuotą srovę su srovės nuostatu ir pagal tai sugeneruoti paklaidos signalą, trečias žingsnis – sustiprinti paklaidos signalą, kad būtų galima sukurti reikiamo dydžio korekcines sroves, ketvirtas žingsnis – pagal korekcines sroves paduoti reikiamo dydžio sroves į variklio apvijas.[15]

FOC metodas, kaip ir sinusinis valdymas, sumažina sukimo momento trūkius, bet taip pat jis pasižymi didesniu efektyvumu esant mažiems ir vidutiniams variklio greičiams negu pastarasis metodas. Tokiam variklio valdymo metodui atitinkamai reikia didelio parametrų matavimo tikslumo bei sudėtingų valdymo algoritmų

1.3. „Sausgyslėmis“ valdomo straublio tipo roboto segmentų konstrukciniai variantai ir „sausgyslių“ skaičius bei išdėstymas segmente

Sausgyslėmis valdomo straublio tipo roboto segmentų galimi konstrukciniai variantai dažniausiai skirstomi į tokias dvi grupes: į segmentus, kurių stuburo ilgiai gali kisti valdant robotą, tai yra sutrumpėti ir pailgėti, ir į segmentus, kurių stuburo ilgiai valdant sausgysles nekinta ir išlieka pastovūs.

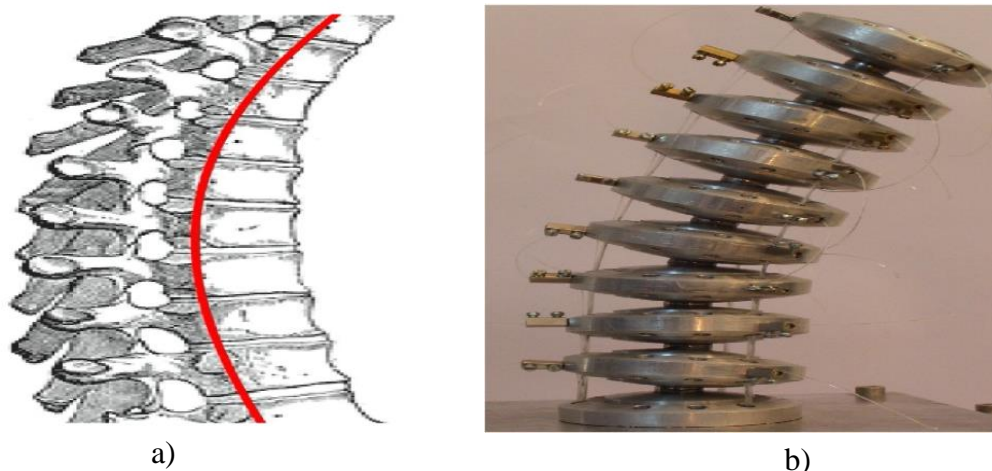
Segmentai, kurių stuburo ilgiai sulenkiant koku nors kampu, išlieka pastovūs, valdymo požiūriu yra laikomi pačiais paprasčiausiais. Tokie segmentai turi 2 laisvės laipsnius, iš kurių vienas laipsnis nusako, jog segmentas gali sukis apie z ašį, o kitas laipsnis nurodo, jog segmentas gali lenktis apie y arba x ašį. Vienas iš galimų variantų segmentui, kurio stuburo ilgis nesikeičia, pagaminti yra tarp atraminių diskų įdėti sferines jungtis, o segmento formai išlaikyti ir stabilumui padidinti tarp atraminių diskų „sausgyslių“ vietose yra įdedamos spyruoklės (žr. 7 pav.). Sferinė jungtis 7 paveikslėlyje pavaizduotame atvejuje yra realizuojama pasitelkiant universalią jungtį, veleną ir rutulinius guolius, taip suteikiant galimybę ne tik atlikti sulenkimo judesį apie y arba x ašis, bet ir nežymius pasukimo ir slenkamuosius judesius tarp dviejų atraminių diskų apie z ašį.[16]



7 pav. Segmento, kurio stuburas yra pastovaus ilgio, struktūrinė schema

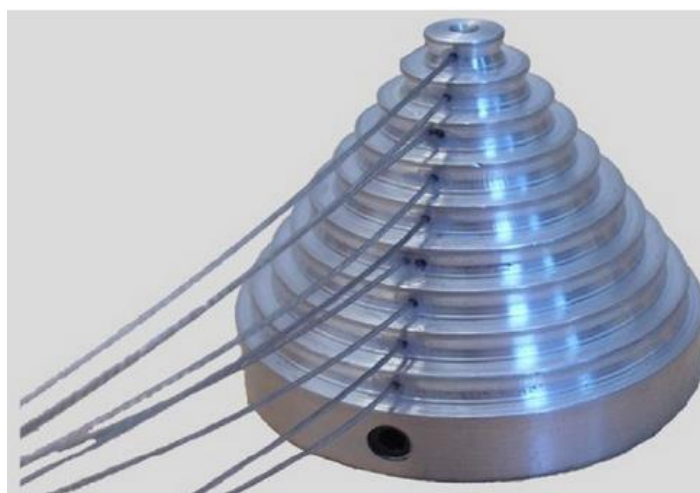
Pasukimo judesys apie z ašį yra labai apribotas dėl spyruoklių įtvirtintų tarp atraminių diskų, o slenkamasis judesys palei z ašį apribotas savo veleno ilgio. 7 paveikslėlyje pavaizduotas roboto segmento konstrukcinis atvejis išsiskiria tuo, jog šio segmento kūrėjai tvirtina, kad esant nežymiesiems leistiniems slenkamiesiems ir sukimo judesiams tarp dviejų universalių jungčių, yra gaunamas šiek tiek didesnis segmento sulenkimo kampas ir geriau pasiskirsto įtempimo jėgos tarp vieno roboto segmento pirmo ir galinio atraminių diskų, nei tarp kiekvieno atraminio disko montuojant universalias jungtis be velenų ir rutulinių guolių. [16]

Dar vienas straublio tipo roboto pastovaus ilgio stuburo segmento konstrukcinis variantas yra įkvėptas ištyrus žmogaus stuburą, jo slankstelius ir jo lankstymąsi. Šiuo atveju, roboto atraminiai diskai yra pagaminami specialios formos - atraminio disko dvi apskritimo formos plokštumos yra išgaubiamos į išorę ir sujungiamos metaliniu rutuliuku (rutuliniu guoliu) (žr. 8 pav.).[17]



8 pav. a) Išlenkto žmogaus stuburo nuotrauka, b) Straublio tipo roboto vieno segmento, kurio stuburas yra pastovaus ilgio, vaizdas

Tokio segmento jungtis vadinama disko-guolio-disko jungtimi (angl. *Disk-Ball-Disk (DBD) joint*). Šiuo atveju, du atraminiai diskai ir guolis, esantis tarp jų, atstoja trumpo ilgio sferinę jungtį, todėl lyginant 7 paveikslėlyje ir 8 paveikslėlyje pavaizduotus roboto segmentų konstravimo variantus, pastarajam variantui pagaminti reikia daug daugiau atraminių diskų, jeigu abiem atvejais roboto segmentų ilgiai turi būti vienodi. Taip pat kiekvienam diskui reikia atskirų siūlų, kuriuos traukiant judinamas atraminis diskas ir taip pat išlaikoma jo pozicija roboto segmento sudėtyje, kad atraminis diskas neiššoktų ir roboto segmentas nesubirtų. Kad vienam roboto segmentui valdyti nereikėtų sumontuoti atskirų elektros variklių, kiekvieno atraminio disko siūlui arba kitaip tariant „sausgyslei“ traukyti, visos segmento „sausgyslės“ yra sugrupuojamos į kelias grupes (8 paveikslėlio b dalyje matomos 3 „sausgyslių“ grupės) ir viena „sausgyslių“ grupė, valdoma su vienu elektros varikliu, pasitelkiant daugiasluoksnį skirtingų skersmenų takų (kūgio formos) skriemulį (žr. 9 pav.).[17]

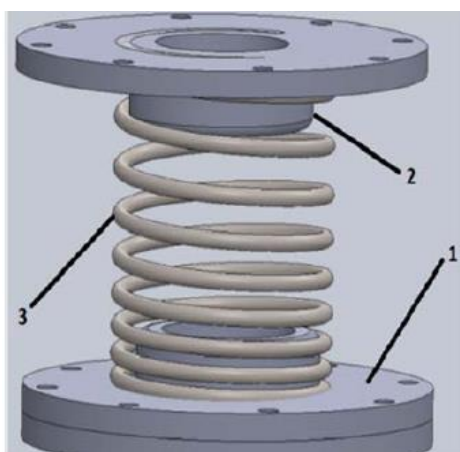


9 pav. Daugiasluoksnis kūgio formos skriemulys

Nors konstruojant straublio tipo roboto segmentą iš disko-guolio-disko jungčių yra sunaudojama daugiau atraminių diskų ir tuo pačiu daugiau konstrukcinių medžiagų, bet šis variantas pasižymi

tvirtesniu roboto segmento stuburu, nei segmentas, pagamintas iš atraminių diskų, spyruoklių ir universalių jungčių.

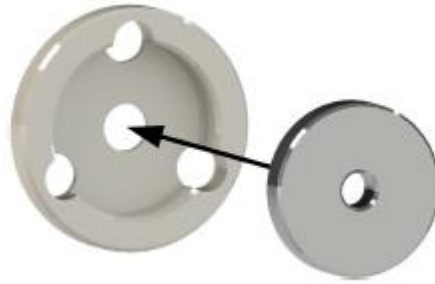
Kita straublio tipo roboto segmento konstrukcinio varianto grupė yra segmentai, kurių stuburas yra kintančio ilgio. Šios grupės segmentai konstrukciniu požiūriu yra mažiau standūs už pastovaus ilgio segmentus, nes jų stuburai gali keisti ilgį, bet dėl tos pačios priežasties jų valdymas yra sudėtingesnis. Straublio tipo robotas, kurio segmento stuburas yra kintamo ilgio, pasižymi didesne darbo zona, negu segmentai, kurių stuburų ilgiai nekinta, nes šie segmentai turi 3 laisvės laipsnius, o pastarieji tik 2 laisvės laipsnius. Šis skirtumas dar akivaizdesnis, kai robotas turi daugiau negu vieną segmentą. Vienas iš dažniausiai sutinkamų straublio tipo roboto kintamo ilgio stuburo segmento konstrukcinių variantų yra pagrįstas spyruoklės taikymu vietoje roboto stuburo, kuris jungia atraminius roboto diskus (žr. 10 pav.).[18]



10 pav. Straublio tipo roboto kintamo ilgio stuburo segmento dalis, kurios stuburas yra spyruoklės pagrindu, 1 – roboto atraminis diskas, 2 – spyruoklės laikiklis, 3 – spyruoklė

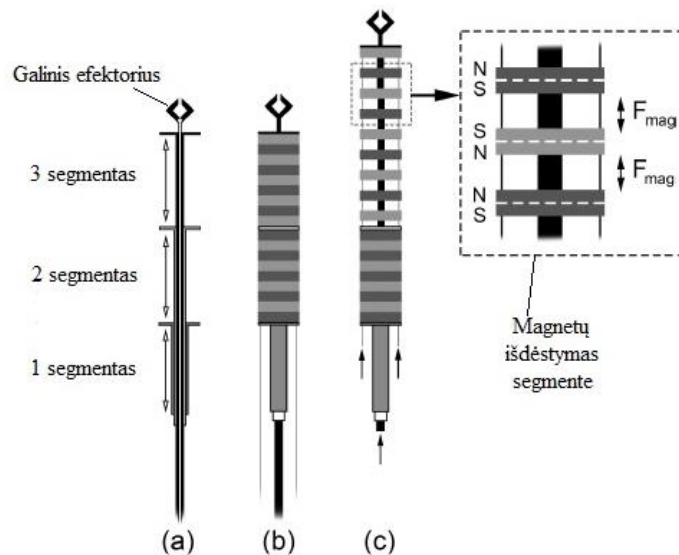
Tokios konstrukcijos segmento, kuris pavaizduotas 10 paveikslėlyje stuburo standumas priklauso nuo spyruoklės standumo. Kuo spyruoklių, laikančių atraminius diskus, standumas mažesnis, tuo sunkiau yra išlaikoma segmento forma jį sulenkus arba segmento forma jam esant pradinėje padėtyje, kuomet jis yra tiesus. Tačiau, įdėjus pernelyg standžias spyruokles, yra susiduriama su tokiomis problemomis, kaip didelės roboto „sausgyslių“ įtempimo jėgos ir nestabilumas. Todėl konstruojant straublio tipo roboto segmentus kintančio stuburo ilgio, kurių stuburas yra padarytas spyruoklės pagrindu, yra siūloma į segmento tarpus tarp atraminių diskų dėti nevienodo standumo spyruokles. Yra siūloma žiūrint nuo segmento pradžios (nuo roboto pagrindo) iki pabaigos (roboto laisvo galo) dėti spyruokles nuo didžiausio standumo, palaipsniui mažinant standumą, iki mažiausio standumo spyruoklių. Jeigu segmente, tarp atraminių diskų yra vis dėlto montuojamos vienodo standumo spyruoklės, tačiau jų standumas yra per mažas, galima įmontuoti į segmentą vientisą spyruoklę kuri eina per visų atraminiuose diskuose padarytas angas (10 paveikslėlyje matomos dvi didelio diametro skylės esančios spyruoklių laikikliuose), taip papildomai sujungiant segmento pradžią su pabaiga ir padidinant bendrą segmento standumą.[18]

Kintančio stuburo ilgio segmentų, kurių stuburas yra spyruoklės pagrindu, turi ilgaamžiškumo problemą. Bėgant laikui, spyruoklės dėvisi ir tuo pačiu praranda savo mechanines savybes. Siekiant apeiti šią problemą ir netaikyti spyruoklių vietoj stuburo, yra siūloma pagaminti atraminius diskus taip, jog juose būtų įmontuoti nuolatiniai magnetai (žr. 11 pav.).



11 pav. Atraminis diskas, kuriame įmontuojamas nuolatinis magnetas

Nuolatiniai magnetai yra sudedami į atraminius diskus ir šie atraminiai diskai užmaunami atsižvelgiant į magnetų poliarumus ant lankstaus, elastinio, teleskopinio (išstumiamo) stuburo, pagaminto iš nikelio ir titano lydinio taip, kad iš eilės einantys atraminiai diskai vienas kitą stumtų su savo magnetiniais laukais (žr. 12 pav.). Segmento paskutinis atraminis diskas yra užtvirtinamas ant segmento stuburo, o pirmasis segmento diskas remiasi į prieš jį esančio segmento įtvirtintą atraminį diską arba į roboto pagrindą. Atraminiai diskai su nuolatiniais magnetais, esančiais tarp pirmo ir trečio atraminių diskų yra paliekami palaidi, nes tarpai tarp šių diskų dėl magnetinių laukų tarpusavio sąveikos pasidaro vienodi. Todėl atraminiams diskams papildomų laikiklių, tokių kaip spyruoklės, nebereikia.[19]



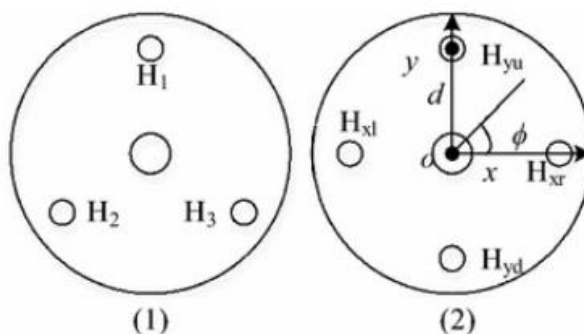
12 pav. Straublio tipo robotas, kurio segmentų stuburas yra kintamo ilgio ir pagaminti pritaikant nuolatinius magnetus, a) 3-jų segmentų lanksčių teleskopinių stuburų išdėstymas, b) Pilnai sutraukti 2-as ir 3-ias segmentai, c) Pilnai sutrauktas 2-as segmentas ir pilnai ištiestas 3-ias segmentas

Segmento sutraukimo metu (kai mažinamas segmento ilgis), segmento stuburas yra patraukiamas į roboto pagrindą. Kadangi segmento stuburas yra vientisas elastinis vamzdelis, negalintis keisti savo ilgio, todėl jis montuojamas robote kaip teleskopinis stuburas ir prireikus yra išstumiamas arba sutraukiamas atgal (žr. 12 pav. (b) ir (c)).[19]

Kintančio stuburo ilgio straublio tipo roboto segmentai, kurių atraminiai diskai pagaminti taikant nuolatinius magnetus, išsiskiria iš kintamo ilgio segmentų, kurių stuburas yra pagamintas spyruoklės pagrindu tuo, jog jų segmentų ilgių kitimo diapazonas yra didesnis negu pastarųjų segmentų. Todėl kintančio stuburo ilgio segmentų su nuolatiniais magnetais atraminuose diskuose darbo zona yra didesnė už segmentų, kurių stuburas yra pagamintas spyruoklės pagrindu.

Bet ši segmento konstrukcija, paremta nuolatiniais magnetais, turi ir savo trūkumą. Didžiausia problema ta, kad neįmanoma pagaminti didelio skersmens segmento, naudojant nuolatinius magnetus atraminiuose diskuose, kadangi didinant atraminius diskus, didėja nuolatinių magnetų dydžiai ir tuo pačiu stiprėja magnetiniai laukai. Todėl atsiranda didelės įtempimo jėgos roboto „sausgyslėse“ valdant segmentą, dėl ko roboto „sausgyslė“ arba, kitaip tariant, viela ar kitokios medžiagos siūlas valdantis segmentą, greičiau dėvisi ir gali greitai nutrūkti. Todėl straublio tipo roboto, kurio stuburas kintamo ilgio, segmentai pagaminti atraminiuose diskuose taikant nuolatinius magnetus yra dažniausiai taikomi medicinoje dėl savo nedidelio segmento ilgio ir skersmenio.

Apsibrėžus kokio tipo segmento stuburas turės būti (kintančio ilgio arba nekintančio) yra sprendžiama, kiek vienam segmentui valdyti reikės „sausgyslių“ arba, kitaip tariant, vielų arba kitokios medžiagos siūlų. To reikia, kad būtų aišku, kiek reikės vykdiklių, pavyzdžiui, elektros variklių vienam segmentui valdyti, kadangi vienai „sausgyslei“ valdyti reikia vieno vykdiklio. „Sausgyslių“ skaičius straublio tipo roboto segmente negali būti mažesnis negu 3 sausgyslės, nes mažiausiai su 3-is „sausgyslėmis“ yra pilnai realizuojami segmento 2 laisvės laipsniai, kad segmentą būtų galima sulenkti apie y arba x ašį ir pasukti apie z ašį. Toliau „sausgyslių“ skaičių galima didinti, bet jis yra apribojamas segmentų atraminių diskų skersmens dydžiu, pačios „sausgyslės“, tai yra, vielos arba kitos medžiagos siūlo storium ir segmentų skaičiumi, iš kurių sudarytas robotas. Kitaip tariant, didelio skaičiaus „sausgyslių“ negalima pasiekti žiūrint iš konstrukcinės pusės, todėl dažniausiai renkamos iš dviejų variantų: pirmas variantas, kai segmentui valdyti yra skiriamos 3 „sausgyslės“, antras variantas, kai segmentui skiriamos 4 „sausgyslės“ (žr. 13 pav.).[5]



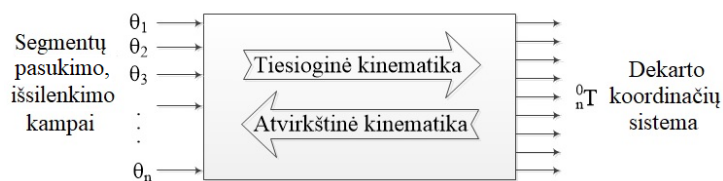
13 pav. Straublio tipo roboto „sausgyslių“ skaičius ir išdėstymas segmento atraminiame diske (vaizdas iš viršaus), 1) 3-jų „sausgyslių“ išdėstymas atraminiame diske, 2) 4-ių „sausgyslių“ išdėstymas atraminiame diske

Straublio tipo robote „sausgyslės“ yra išdėstomos atraminiuose diskuose palei disko kraštą apskritimu vienodais kampais. Tai reiškia, jeigu segmentui valdyti reikia 3 „sausgyslių“ tai to segmento kiekviename atraminiame diske turi būti padarytos angos „sausgyslei“ prakišti kas 120° .

Kuo yra didesnis „sausgyslių“ skaičius, skirtas vienam segmentui valdyti, tuo geriau yra paskirstomi segmento „sausgyslių“ įtempimai roboto segmento valdymo metu. Bet esant dideliame „sausgyslių“ skaičiui vienam segmente, lieka vis mažiau vietos sekančio segmento „sausgyslėms“ ir dėl to tenka didinti roboto segmento atraminių diskų skersmenis. Be to, didėja ir vykdiklių skaičius, skirtas valdyti vienam segmentui, dėl ko mažėja roboto ekonomiškas ir efektyvumas. Todėl dažniausiai ir yra naudojamos 3-jų arba 4-ių roboto „sausgyslių“ kombinacijos vienam segmentui valdyti, projektuojant didesnės galios straublio tipo robotus toliau svarstoma, kiek reiktų didinti šį „sausgyslių“ skaičių kiekvienam segmentui atskirai.

1.4. Straublio tipo roboto tiesioginės ir atvirkštinės kinematikos uždaviniai

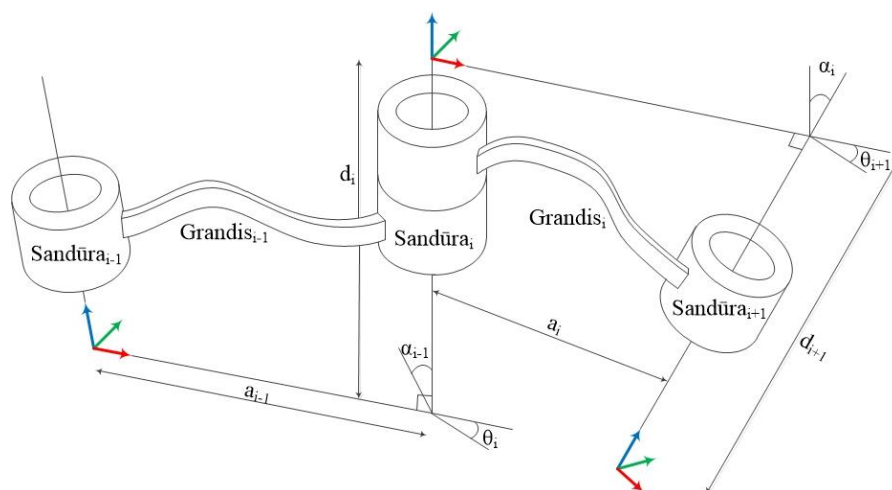
Vienas iš pagrindinių tiriamojo darbo uždavinių yra suvaldyti daugiau negu vieną segmentą turintį straublio tipo robotą taip, kad jo galinis atraminis diskas (žr. 1 pav.) pasiektų iš anksto nurodytą koordinatę ir galinio segmento galo krypties vektorius (žr. 1 paveikslėlyje, 2-o roboto segmento galinio disko plokštumai statmenas vektorius) būtų lygus iš anksto užduotam krypties vektoriui. Norint išspręsti šį uždavinį, reikia rasti roboto kiekvieno segmento bendruosius parametrus tam, kad būtų aišku kiek ir kaip reikia sulenkti roboto kiekvieną segmentą, kad būtų pasiektos iš anksto užbrėžtos koordinatės ir norima roboto galo kryptis. Tai yra kinematikos uždavinys, kurį galima spręsti vienu iš dviejų būdų (žr. 14 pav.): tiesioginiu būdu, kitaip tariant, kaip tiesioginės kinematikos (angl. *Forward kinematics*) uždavinį arba atvirkštiniu būdu, kaip atvirkštinės kinematikos (angl. *Inverse kinematics*) uždavinį.



14 pav. Tiesioginės ir atvirkštinės kinematikos uždavinių sprendimo kryptys

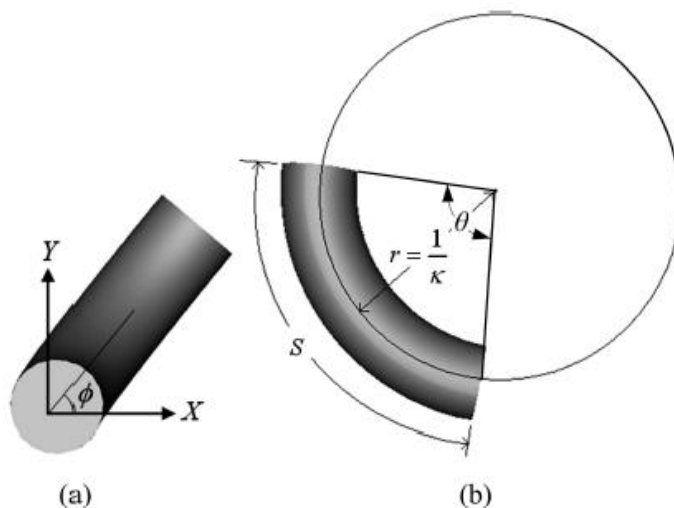
1.4.1. Pagrindiniai straublio tipo roboto segmento parametrai ir Oilerio kampai

Sprendžiant bet kokio tipo roboto kinematikos uždavinį pirmiausiai yra svarbu išsiaiškinti, kaip matematiškai reikia aprašyti kiekvieną roboto dalį, kuri, judinama erdvėje, pakeičia savo poziciją erdvėje ir kaip šias dalis matematiškai tarpusavyje susieti. Tipinis stacionarus robotas manipulatorius yra sudarytas iš standžių, nelanksčių grandžių, kurias jungia lanksčios jungtys (ašys), tai yra kinematinės poros. 1955 metais mokslininkai Denavit ir Hartenbergas įrodė, kad dviejų tarpusavyje sujungtų roboto grandžių pozicijų pokyčiui tarp sandūrų aprašyti reikalingi 4 parametrai (žr. 15 pav.), dar vadinami *Denavito-Hartenbergo* parametrais: grandies ilgis a_{i-1} , grandies aukštis d_i , grandies pasisukimas α_{i-1} ir sandūros kampas θ_i . [20]



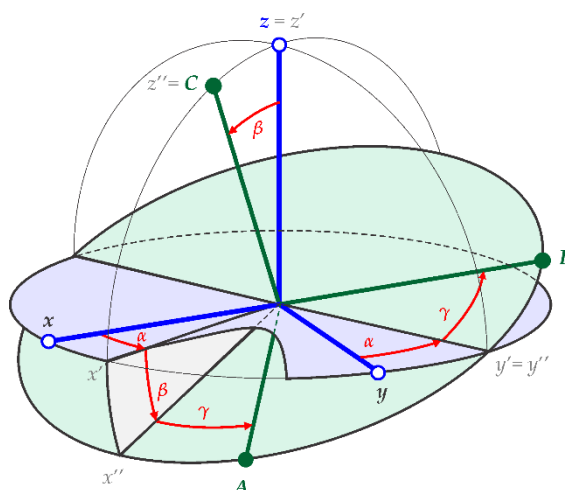
15 pav. Roboto, turinčio dvi grandis ir sukamąją jungtį (sandūrą) brėžinys su *Denavito-Hartenbergo* parametrais

Straublio tipo roboto atveju standartiniais *Denavito-Hartenbergo* parametrais roboto segmentų negalima aprašyti, nes straublio tipo robotas neturi slankiojančių arba sukamųjų jungčių, kaip pavaizduota 15-jame paveikslėlyje, o ir pats roboto segmentas yra lankstus. Todėl straublio tipo roboto segmentas turi būti aprašomas specifiniais parametrais. Žinant, jog valdant straublio tipo roboto bet kokios galimos konstrukcijos segmentą, jis yra sulenkiamas taip, jog segmento forma primena tam tikro skersmens apskritimo, lanko formą yra teigiama, jog šis apskritimo lankas yra taisyklingos formos. Remiantis tuo, kiekvieną straublio tipo roboto segmentą galima aprašyti tokiais bendraisiais 3-is segmento parametrais (žr. 16 pav.): roboto stuburo ilgiu – s , roboto stuburo išlenkimu – k ir roboto stuburo pasisukimo kampu – ϕ . [21]



16 pav. Straublio tipo roboto bendrieji segmento parametrai a) Straublio tipo roboto vieno segmento vaizdas iš apačios, b) Straublio tipo roboto vieno segmento vaizdas iš šono

Roboto stuburo ilgio parametras s padeda apibrėžti atstumą nuo segmento pradžios iki galo. Priklausomai nuo segmento konstrukcijos, tai yra, nuo to ar segmentas yra pastovaus ilgio ar kintančio ilgio, šis parametras taip pat atitinkamai gali būti konstanta arba kintantis dydis. Su roboto stuburo išsilenkimo parametru k per apskritimo lanko kampo skaičiavimo formulę yra išreiškiamas sulenkimo kampas θ , šiuo atveju, apie y ašį (žr. 16 pav.). Roboto stuburo pasisukimo kampu ϕ yra aprašomas segmento pasukimo kampas x ir y plokštumoje apie z ašį. Žinant visus šiuos straublio tipo roboto parametrus, galima pilnai nustatyti roboto kiekvieno segmento padėtį trimatėje erdvėje, pritaikant Oilerio kampus (angl. *Euler angles*) ϕ , θ , ψ , kuriais apibūdinama taško padėtis nejudamosios stačiakampės Dekarto koordinatinių sistemos $Oxyz$ atžvilgiu (žr. 17 pav.). [22]



17 pav. ZYZ Oilerio kampai Dekarto koordinatinių sistemoje

Kiekvienas iš Oilerio kampų ϕ , θ , ψ išreiškia pasukimo kampą apie kurią nors trimatės koordinatinių sistemos ašį. Visi trys Oilerio kampai ϕ , θ , ψ sudaro Oilerio kampų sistemą. Iš viso yra 12 galimų Oilerio kampų sistemų: z-x-z, x-y-x, y-z-y, z-y-z, x-z-x, y-x-y, x-y-z, y-z-x, z-x-y, x-z-y, z-y-x ir y-x-z. Kiekvienas Oilerio sistemos pavadinimas apibūdina apie kokias ašis kokiais kampais yra nuosekliai sukamas taškas Dekarto koordinatinių sistemoje, pavyzdžiui z-x-z reiškia, jog pirmiausiai taškas pasukamas kampu ϕ (α) apie z ašį, tada taškas pasukamas kampu θ (β) apie pasuktąją ašį x' ir galiausiai taškas pasukamas kampu ψ (γ) apie pasuktąją z'' ašį.[23]

Dažniausiai robotikoje yra taikomos tokios 3 Oilerio kampų sistemos z-x-z, z-y-z ir x-y-z, bet straublio tipo roboto atveju, tinkamiausia Oilerio kampų sistema z-y-z (ZYZ) (žr. 17 pav.). Pritaikant ZYZ Oilerio kampų sistemą straublio tipo roboto segmento galo taško padėčiai Dekarto koordinatinių sistemoje išreikšti, kai bendrieji roboto segmento parametrai yra s , k , ϕ yra laikoma, jog Oilerio kampas ϕ (α) yra nenaudojamas, todėl yra lygus 0, Oilerio kampas θ (β) lygus segmento stuburo pasisukimo kampui – ϕ ir paskutinis Oilerio kampas ψ (γ) yra lygus segmento stuburo sulenkimo kampui θ , kuris gaunamas iš segmento išlenkimo parametro k . Žinant roboto kiekvieno segmento bendruosius parametrus bei ZYZ Oilerio kampus, galima atvaizduoti ir tiksliai žinoti visų segmentų padėtis ir orientacijas trimatėje koordinatinių sistemos erdvėje.[22]

1.4.2. Straublio tipo roboto tiesioginės kinematikos uždavinys ir jo sprendimo metodai

Ieškant straublio tipo roboto kiekvieno segmento parametrų yra sprendžiamas kinematikos uždavinys, šiame poskyryje apžvelgsime tiesioginės kinematikos uždavinį ir jo sprendimo metodą straublio tipo robotui. Tiesioginės kinematikos uždavinio sprendimo principas yra toks – nurodant roboto segmento pasukimo ir sulenkimo kampus (pvz. ZYZ Oilerio kampai), ieškoma roboto galinio taško padėtis Dekarto koordinatinių sistemoje (žr. 14 pav.). Kitaip tariant, kiekvienas roboto segmentas yra pasukamas koku nors kampu ir yra ieškomos roboto galinio segmento galo taško koordinatės.

Ieškant straublio tipo roboto segmentų bendrųjų parametrų yra siūloma taikyti modifikuotą *Denavito-Hartenbergo* metodą (angl. *modified Denavit-Hartenberg procedure*), dar vadinamą *Denavito-Hartenbergo* modifikuotų parametrų metodu. Toks pavadinimas kilo todėl, kad segmentų parametrus rasti, jie yra išreiškiami pakeistais (modifikuotais) *Denavito-Hartenbergo* parametrais, kurie yra aprašyti 2.1 poskyryje.[24]

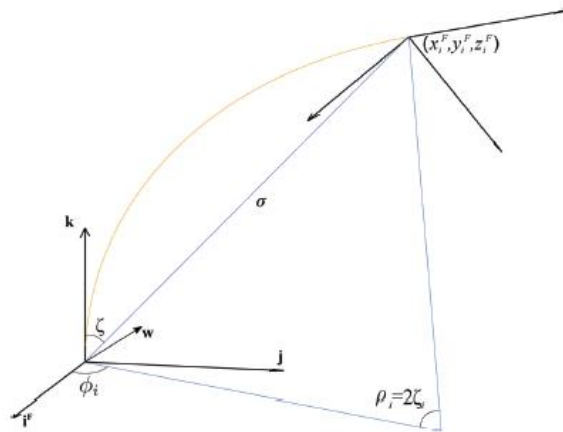
Standartinis *Denavito-Hartenbergo* parametrų metodas netinka straublio tipo roboto tiesioginės kinematikos uždaviniui spręsti, nes straublio tipo robotas neturi slankiojančių arba sukamųjų jungčių, kurios jungia standžias roboto grandis tokias, kurios yra būdingos tipiniams n ašių turinčioms robotinėms rankoms – manipulatoriams. Todėl siekiant pritaikyti *Denavito-Hartenbergo* parametrų metodą straublio tipo robotui, pirmiausia daroma prielaida, jog kiekvienas roboto segmentas lankstosi taip, jog visada yra išlaikoma apskritimo lanko forma, o ne elipsė. Šie apskritimo lankai yra sujungiami į nuoseklia grandinę, kuri yra laikoma straublio tipo roboto stuburu. Kiekvienas roboto segmentas aprašytas bendraisiais parametrais: roboto stuburo ilgiu - s , roboto stuburo išlenkimu – k ir roboto stuburo pasisukimo kampu – ϕ . [24]

Kadangi standartinė *Denavito-Hartenbergo* parametrų matrica yra sudaryta iš pasukimų kampų θ ir objektų poslinkių erdvėje atstumų, kurie lygūs d_i , straublio tipo roboto segmentų parametrai su *Denavito-Hartenbergo* parametrų matrica yra susiejami tokia matematine išraiška:

$$[\underline{\theta} \quad \underline{d}]^T = \underline{f}_1(s, k, \phi) \quad (1)$$

Roboto segmento parametrai s , k ir ϕ išreiškiami kaip funkcija f_1 , kuri yra naudojama *Denavito-Hartenbergo* parametru matricoje vietoje standartinių *Denavito-Hartenbergo* parametru $A(\underline{f}_1(s, k, \phi)) = A(s, k, \phi)$. Taip modifikuojami *Denavito-Hartenbergo* parametrai ir straublio tipo roboto forma yra susiejama su roboto galinio atraminio disko taško koordinatėmis. Toliau skaičiavimams pasitelkiamos posūkio matricos, taip matematiškai roboto segmentai yra lenkiami, bei sukami apie y ir z ašis.[24]

Tiesioginės kinematikos uždavinį dar yra siūloma spręsti ir kitaip, pirmiausia pasirenkant kitokius segmento pasukimo, sulenkimo kampus bei segmento stuburo ilgio apibrėžiančius parametrus ir juos iš karto išreikšti funkcijomis, kuriose yra taikomos Dekarto koordinatinių sistemos koordinatės. Šiuo atveju segmentas yra apibrėžiamas vektoriaus ilgiu σ_i tarp segmento pradžios taško ir galinio taško, kampu ζ_i tarp šio vektoriaus ir z ašies ir segmento pasukimo kampu ϕ_i apie z ašį (žr. 18 pav.).[25]



18 pav. Straublio tipo roboto segmento parametrai

Toliau segmento galo taško koordinatė p_i^F yra išreiškiama per formulę (2). Keičiant parametru reikšmes σ_i , ζ_i , ϕ_i yra tiesiogiai ieškomos roboto segmento galinio taško koordinatės, nenaudojant ZYZ Oilerio kampų. Vis dėl to, jeigu yra ieškoma ir ZYZ Oilerio kampų straublio tipo roboto segmentui, išreikšus pasukimo apie z ašį kampą ϕ_i kaip vektorių w_i (3), o segmento sulenkimo kampą kaip ρ_i (4) ir jiems pritaikius Rodrigeso formulę (angl. *Rodrigues' formula*) gaunama posūkio matrica R_i . Šią gautą posūkio matricą R_i , sudauginus su segmento galinio taško funkcija p_i^F , yra gaunama homogeninė transformacijų matrica T_i (5), kurios pagalba jau galima rasti ZYZ Oilerio kampus.[25]

$$p_i^F = [\sigma_i \sin \zeta_i \cos \phi_i, \quad \sigma_i \sin \zeta_i \sin \phi_i, \quad \sigma_i \cos \zeta_i] \quad (2)$$

$$w_i = [-\sin \phi_i, \quad \cos \phi_i, \quad 0] \quad (3)$$

$$\rho_i = 2\zeta_i \quad (4)$$

$$T_i = \begin{bmatrix} (s\phi_i)^2(1 - c2\zeta_i) + c2\zeta_i & s\phi_i c\phi_i(c2\zeta_i - 1) & c\phi_i s2\zeta_i & \sigma_i s\zeta_i c\phi_i \\ s\phi_i c\phi_i(c2\zeta_i - 1) & (c\phi_i)^2(1 - c2\zeta_i) + c2\zeta_i & s\phi_i s2\zeta_i & \sigma_i s\zeta_i s\phi_i \\ -c\phi_i s2\zeta_i & -s\phi_i s2\zeta_i & c2\zeta_i & \sigma_i c\zeta_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

Nors ir tiesioginės kinematikos uždavinys roboto segmentų išsilenkimo ir pasukimo kampams rasti sprendžiamas pritaikius modifikuotą *Denavito-Hartenbergo* metodą, bet ieškoti kampų sprendžiant kinematikos uždavinį tiesioginiu metodu, kai roboto galinis segmentas turi pasiekti iš anksto trimatėje erdvėje nurodyto taško koordinatės ir galinio segmento galo kryptis turi sutapti su iš anksto nurodyto vektoriaus einančio iš užduoties taško kryptimi, yra gana sudėtinga. Norimam rezultatui pasiekti šį uždavinį sprendžiant tiesioginiu metodu tektų pakartoti labai daug kartų, taip atliekant daug skaičiavimų bei pritaikyti kokį nors paieškos metodą, kuris padėtų artėti prie ieškomo atsakymo. Pritaikius aprašytą antrąjį tiesioginės kinematikos metodą, yra parenkami šiek tiek kitokie segmento parametrai ir išreiškiami funkcijomis σ_i , ζ_i , ϕ_i , segmento ZYZ Oilerio kampams rasti, kai segmento galas turi pasiekti iš anksto nurodyto taško koordinatės. Bet vėlgi, daugiau negu vieną segmentą turinčio straublio tipo roboto tiesioginį kinematikos uždavinį pastaruoju metodu reikia spręsti daugybe kartų. Taip pat, visus roboto segmentus reikia susieti, nes aprašytas metodas iš eilės einančių segmentų nesusieja. Tai siūloma padaryti papildomai panaudojant Jakobiano matricas (angl. *Jacobian matrix*), bet dėl to šis skaičiavimo metodas pasidaro labai sudėtingas ir jame vis tiek iš anksto į sąlygą nėra įtraukta roboto galinio segmento galo orientaciją nurodantis vektorius.

1.4.3. Straublio tipo roboto atvirkštinės kinematikos uždavinys ir jo sprendimo metodai

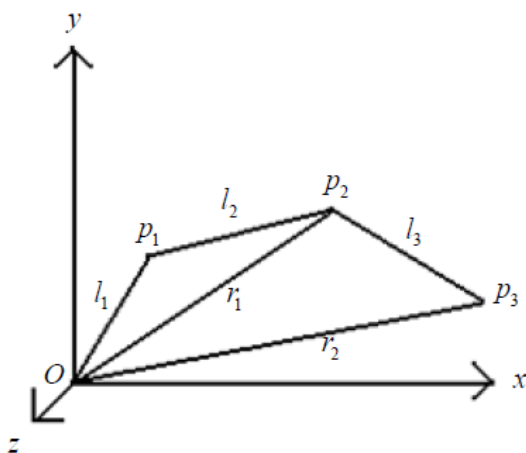
Atvirkštinės kinematikos uždavinio sprendimo kryptis yra priešinga tiesioginės kinematikos uždavinio sprendimo kryptčiai (žr. 14 pav.). Sprendžiant atvirkštinį kinematikos uždavinį yra turimos roboto galo taško koordinatės, kai robotas jau yra pasiekęs nurodytą poziciją ir yra ieškomi kampai, kuriais turi būti pasukamos arba perstumiamos roboto grandys. Straublio tipo roboto atveju yra iš karto ieškoma kampų, kuriais turi būti sulenkta apie y ašį ir pasuktas apie z ašį kiekvienas roboto segmentas, norint pasiekti nurodyto taško koordinatės. Šiuos pasukimo ir sulenkimo kampas dar galima gauti iš pagrindinių roboto segmentų parametrų.

Vienas iš galimų būdų, sprendžiant straublio tipo roboto atvirkštinį kinematikos uždavinį, yra siūloma pritaikyti geometrinį sprendimo metodą. Sprendžiant atvirkštinį kinematikos uždavinį geometrinio metodu uždavinys yra skeliamas į dvi dalis: pirmiausia, ieškomos roboto kiekvieno segmento galinio taško koordinatės, o po to, žinant šias koordinatės, ieškomi kiekvieno segmento pagrindiniai parametrai s , k ir ϕ . [26]

Iš anksto yra žinomos dviejų taškų koordinatės, tai roboto pirmo segmento pradžios taško koordinatė, kuri trimatėje erdvėje visada yra lygi $(0, 0, 0)$ ir nurodomas užduoties taškas (x, y, z) , kurią reikia pasiekti galiniam roboto segmentui. Kai straublio tipo robotą sudaro tik vienas segmentas, atvirkštinis kinematikos uždavinys geometrinio metodu yra išsprendžiamas labai greitai, nes pirmos skaičiavimo dalies nebereikia atlikti ir iš karto yra ieškomi segmento pagrindiniai parametrai. Bet kai robotą sudaro daugiau negu vienas segmentas, pavyzdžiui, trys segmentai, segmentų, kurių galiniai taškai yra nežinomi, yra ieškomi piešiant koordinačių sistemoje du trikampus (žr. 19 pav.). Šie du trikampiai yra brėžiami taip, kad pirmasis trikampis ΔOP_1P_2 jungtų pirmo segmento pradinį tašką O su jo galiniu tašku P_1 ir antro segmento pradinį tašką P_1 su jo galiniu tašku P_2 bei tuo pačiu su pirmo segmento pradžios tašku O , o antrasis trikampis ΔOP_2P_3 sujungia antro segmento galinį tašką P_2 (trečio segmento pradinį tašką) su trečio segmento galiniu tašku P_3 . Turint šiuos trikampus toliau yra laisvai pasirenkami trikampių kraštinių ilgiai, kurie turi tenkinti šias nelygybes [26]:

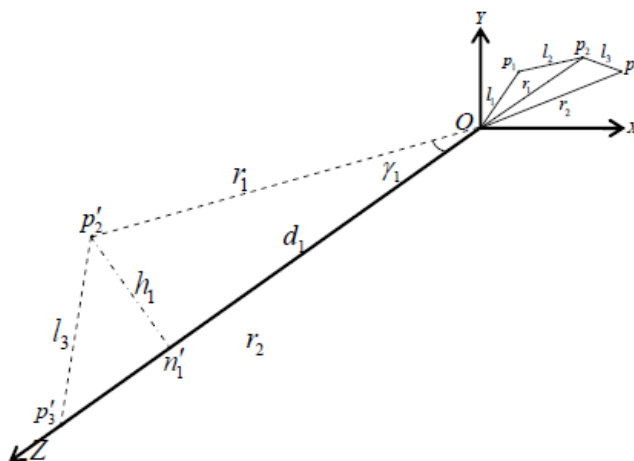
$$r_2 + l_3 \geq r_1 \geq |r_2 - l_3| \quad (6)$$

$$l_1 + l_2 \geq r_1 \geq |l_1 - l_2|$$



19 pav. Straublio tipo roboto 3-jų segmentų pradiniai ir galiniai taškai sujungti į trikampius

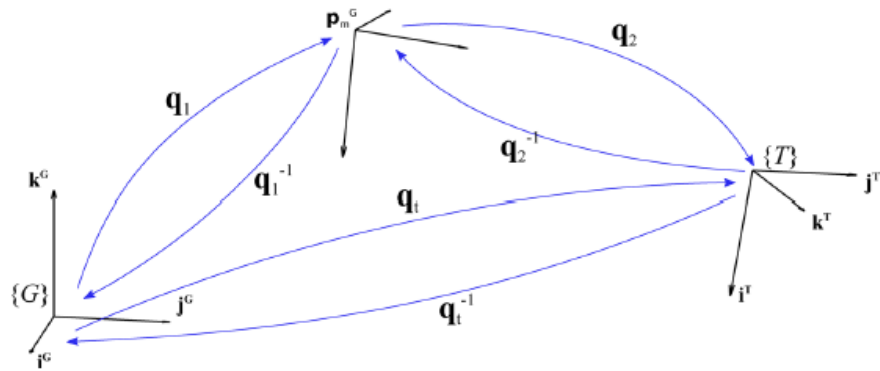
Parinkus abiejų trikampių ΔOP_1P_2 ir ΔOP_2P_3 visų kraštinių ilgius, toliau yra ieškomi taškų P_1 ir P_2 pasukimo kampai θ apie trikampių įžambines. Pirmiausia, ΔOP_2P_3 trikampis pasukamas, pasukant P_3 tašką į z ašį taikant posūkio matricas (žr. 20 pav.). Tada šio pasukto trikampio $\Delta OP_2'P_3'$ taškas P_2' pasukamas laisvai pasirenkamu kampu θ apie z ašį ir tas pats pakartojama su trikampiu ΔOP_1P_2 . Žinant trikampių kraštinių ilgius ir taškų pasukimus apie z ašis yra gaunamos taškų P_1 ir P_2 nežinomos koordinatės ir toliau jau ieškomi segmentų pagrindiniai parametrai s , k ir ϕ . [26]



20 pav. ΔOP_2P_3 trikampio įžambinė pasukta į z ašį

Sprendžiant atvirkštinį kinematikos uždavinį geometriniais metodais yra teigiama, jog šis metodas yra tinkamas valdyti 3-jų segmentų robotą, kurio segmentų stuburas yra kintančio ilgio, realiu laiku dėl valdymo ciklo trumpo skaičiavimo laiko (3 ms). Bet šiame metode apibrėžus tašką, kurį reikia pasiekti koordinatėmis, nėra apibrėžta trečio segmento orientavimo sąlyga, kuri valdant kokį nors įrankį pritaistytą prie paskutinio segmento laisvo galo, neretai yra labai svarbi. [26]

Dar vienas atvirkštinio kinematikos uždavinio sprendimo metodas yra kvarterionų pritaikymas robotų segmentų orientacijoms erdvėje surasti. Šio metodo atveju yra pakeičiami segmentų pagrindiniai parametrai iš s , k ir ϕ į σ_i , ζ_i , ϕ_i , kurie yra išreikšti funkcijomis, apibrėžiančiomis segmento pasukimą ir išlenkimą trimatėje Dekarto koordinatinių sistemoje (žr. 1.4.2 skyrelyje aprašytą antrąjį tiesioginės kinematikos sprendimo metodą). 1.4.2 skyrelyje aprašytas antrasis tiesioginės kinematikos sprendimo metodas turėjo trūkumą, jog robotų segmentai nėra tarpusavyje matematiškai susieti, bet pritaikius kvarterionus šiems σ_i , ζ_i , ϕ_i parametrams galima apibrėžti segmentų orientacijas ir pagal jas susieti atskirus segmentus (žr. 21 pav.). [25]



21 pav. Straublio tipo roboto dviejų segmentų orientacijos, išreikštos kvarterionais

Kiekvieno segmento kvarterionas yra išreiškiamas tokia formule (7), kuri vėliau perdaroma į (8) taip, jog iš karto būtų galima įrašyti kiekvieno segmento galinio taško koordinatas x_i^F, y_i^F, z_i^F (žr. 18 pav.) ir gauti segmento orientaciją išreiškiantį kvarterioną. Šio metodo kūrėjai tvirtina, jog kvarterionais susiejus roboto segmentus, galima išgauti visus galimus teisingus roboto segmentų išsilenkimo variantus roboto darbo zonoje.[25]

$$q_i = \cos \frac{\rho_i}{2} + w_{ii} \sin \frac{\rho_i}{2} i^F + w_{ij} \sin \frac{\rho_i}{2} j^F + w_{ik} \sin \frac{\rho_i}{2} k^F \quad (7)$$

$$q_i = \frac{z_i^F - y_i^F i^F + x_i^F j^F}{\sqrt{(x_i^F)^2 + (y_i^F)^2 + (z_i^F)^2}} \quad (8)$$

Lyginant pirmą aprašytą atvirkštinės kinematikos uždavinio sprendimo geometrinį metodą su antruoju aprašytu kvarterionų metodu galima teigti, jog pirmas metodas yra greitas ir paprastas, bet su juo yra sunkiau pasiekama norima galinio segmento orientacija. Kadangi geometriniam metode nėra paminėta, kaip galima išreikšti galinio segmento orientaciją, kurią po to būtų galima susieti su vektoriumi, nurodančio norimą segmento orientavimo kryptį. Antrasis aprašytas atvirkštinio kinematikos uždavinio sprendimo metodas paremtas kvarterionų taikymu šią krypties nurodymo problemą išsprendžia, bet abu aprašyti atvirkštinės kinematikos uždavinio sprendimo metodai tinkami straublio tipo robotams, kurių segmentų stuburų ilgiai yra ne pastovūs, o kintantys.

2. Straublio tipo roboto valdymo algoritmo kūrimo metodika

2.1. Straublio tipo roboto valdymo algoritmą apibrėžiančios sąlygos

Šio tiriamojo darbo tikslas yra sukurti algoritmą, skirtą valdyti straublio tipo robotą. Išanalizavus galimus straublio tipo konstrukcijos variantus bei roboto kinematikos uždavinius, buvo apibrėžtos sąlygos, kurių laikantis reikia kurti valdymo algoritmą.

Pirma sąlyga valdymo algoritmui – valdomas robotas turi būti sudarytas iš 3-jų segmentų.

Antra sąlyga valdymo algoritmui – vykdant skaičiavimus, roboto visų segmentų stuburų ilgiai turi būti pastovaus, iš anksto nurodyto ilgio (nurodyta apžvelgus galimus segmentų konstrukcinius variantus žr. 1.3 poskyrį) bei jų, kiekvieno segmento atskirai, stuburų ilgius privaloma leisti pasirinkti norimo ilgio.

Trečia sąlyga valdymo algoritmui – roboto kiekvieno segmento sulenkimo kampai apie x arba y ašis turi būti iš anksto apribojami užduoties sąlygoje (nurodyta remiantis tuo, jog segmentų sulenkimo kampai yra apriboti dėl pačios segmentų konstrukcijos, žr. 1.1 poskyrį). Segmento didžiausias sulenkimo kampas gali būti lygus 120°.

Ketvirta sąlyga valdymo algoritmui – taškas, kurį norima, jog pasiektų roboto 3-io segmento galas bei 3-io segmento galo orientaciją išreiškiantis vienetinis vektorius turi būti nurodomi užduoties sąlygoje prieš pradėdant atlikti skaičiavimus.

Penkta sąlyga valdymo algoritmui – algoritmo apskaičiuoti segmento parametrai turi būti pritaikomi roboto segmentų „sausgyslių“, kitaip tariant, segmentus valdančių vielų, trosų ar siūlų, ilgiams rasti.

Remiantis šiomis penkiomis išvardytomis sąlygomis, toliau buvo kuriamas straublio tipo roboto valdymo algoritmas.

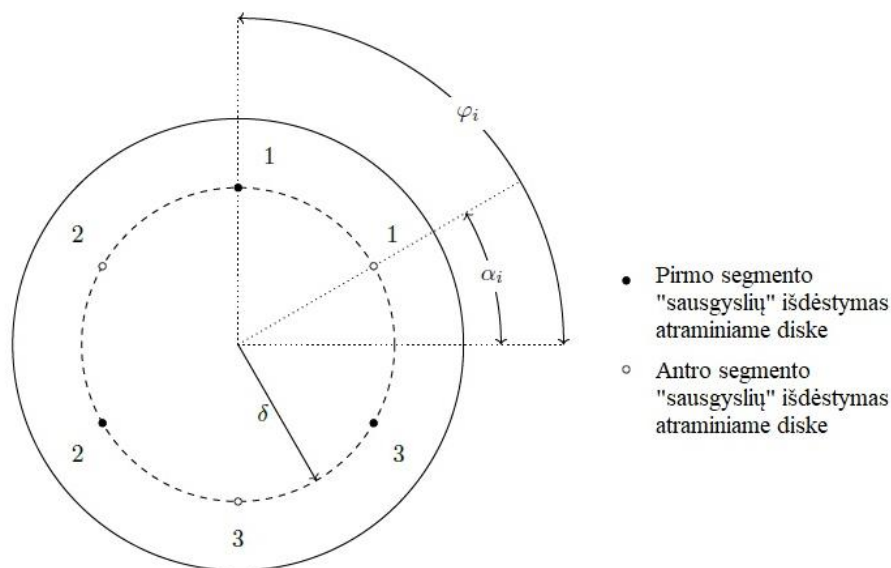
2.2. Straublio tipo roboto segmentų „sausgyslių“ ilgių skaičiavimo formulė ir valdymo algoritmo ieškomų parametrų apibrėžimas

Kuriamo straublio tipo valdymo algoritmo tikslas yra surasti tokius segmentų parametrus, kurie po to būtų taikomi roboto segmentų „sausgyslių“ ilgiams skaičiuoti. Siekiant užtikrinti, jog ieškomi segmentų parametrai bus tinkami šiam tikslui įgyvendinti, reikia išanalizuoti universalią segmentų „sausgyslių“ ilgių skaičiavimo formulę, tinkančią visoms straublio tipo roboto konstrukcijoms, kurių segmentų stuburai yra pastovaus ilgio ir „sausgyslių“ išdėstymai atraminiuose diskuose yra vienodi. Remiantis literatūros šaltiniais [27][28], universali formulė roboto „sausgyslių“ ilgiams skaičiuoti yra tokia (9):

$$l_i = l(1 - k\delta \cos(\varphi_i - \phi)) \quad (9)$$

čia l_i – roboto segmento i -sios „sausgyslės“ ilgis, l – roboto segmento stuburo ilgis, k – roboto segmento išsilenkimą apibūdinanti konstanta, δ – atstumas roboto atraminio disko plokštumoje nuo roboto stuburo centro taško iki roboto „sausgyslės“ centro taško, φ_i – roboto i -osios „sausgyslės“ centro taško pasukimo apie z ašį kampas atraminiame diske, ϕ – roboto segmento pasukimo apie z ašį kampas.

Pagal 2.1 poskyryje nurodytas 5 sąlygas, apibrėžiančias straublio tipo algoritmo kūrimą iš (9) formulės kintamųjų yra žinomi štai šie: l – kadangi jis apibrėžiamas uždavinio sąlygos pradžioje ir jis nekinta, δ ir φ_i – kurie yra apibrėžti, kai yra kuriama roboto segmento konstrukcija (žr. 22 pav.).



22 pav. Straublio tipo roboto vieno segmento turinčio 3 „sausgysles“ atraminio disko vaizdas iš viršaus su „sausgyslių“ pozicijų taškus apibūdinančiais parametrais

Išsiaiškinus (9) formulės žinomus kintamuosius tampa aišku, jog nežinomieji kintamieji yra šiuo atveju tik du, tai yra, k ir ϕ , iš kurių k yra išreiškiama (10) formule (žr. 16 pav. b).

$$k = \frac{1}{r} \quad (10)$$

čia k – roboto segmento (apskritimo lanko forma) išsilenkimą apibūdinanti konstanta, r – apskritimo spindulio ilgis.

Nežinomų kintamųjų paieškos skaičiavimo patogumui, k parametą pakeisime į kitą parametą, kuris segmento išsilenkimą parodo laipsniais. Kadangi yra laikoma, jog segmento stuburas yra išlinkęs taip, jog jo forma yra taisyklingo apskritimo lanko formos, išsireiškus stuburo ilgį l su formule (11) ir apskritimo lanko spindulį išreiškus per (10) formulę, gaunam roboto segmento išlenkimą išreikštą per kampą θ (12).

$$l = r\theta \quad (11)$$

$$\theta = lk \quad (12)$$

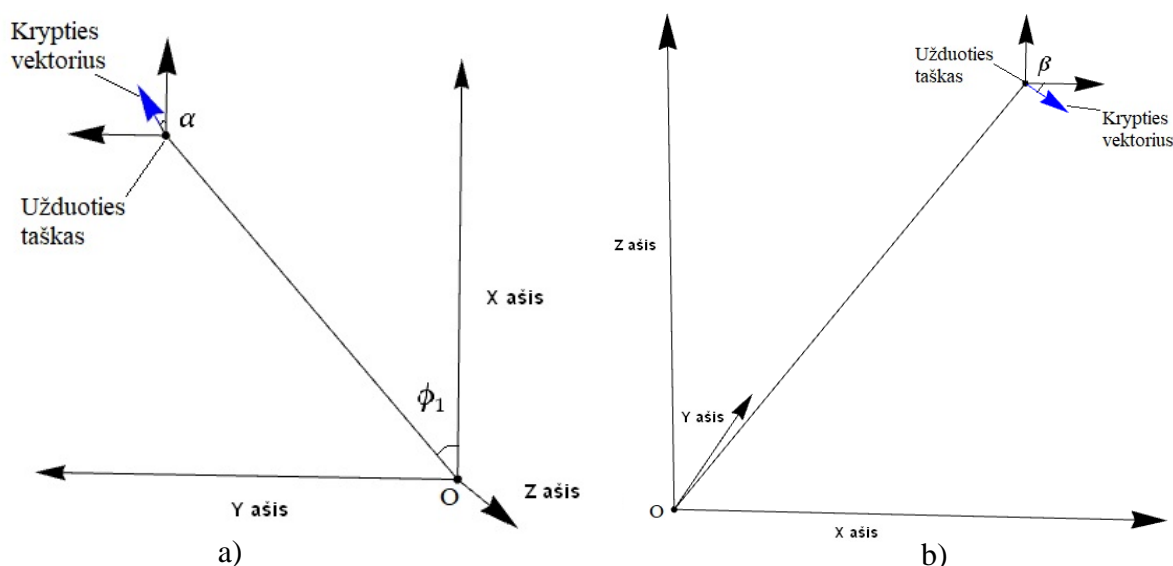
čia l – roboto segmento stuburo (apskritimo lanko forma) ilgis, r – apskritimo spindulio ilgis, θ – apskritimo lanko kampas, k – roboto segmento (apskritimo lanko forma) išsilenkimą apibūdinanti konstanta.

Taigi, kuriamo straublio tipo roboto valdymo algoritmas, žinant taško, kurį reikia pasiekti trečiam roboto segmentui, koordinates ir iš anksto apibrėžus trečio segmento galo kryptį su vienetiniu vektoriumi, turi rasti po du kampus (ϕ , θ) kiekvienam segmentui. Kampas ϕ šiuo atveju apibrėžia segmento pasukimo kampą apie z ašį, o kampas θ apibrėžia segmento sulenkimo kampą apie y ašį. Kitaip tariant, roboto segmento orientaciją išreiškiama ZYZ Oilerio kampais ϕ , θ , ψ (žr. 1.4.1 skyrelį), iš kurių kampas ϕ nenaudojamas ir paliekamas lygus 0° .

2.3. Straublio tipo roboto valdymo algoritmo kūrimo pradžia pritaikant vektorius

Apibrėžus 2.2 poskyryje kintamuosius, kuriuos roboto valdymo algoritmui reikia surasti, tai yra, surasti kampą ϕ , reiškiantį segmento pasukimo kampą apie z ašį ir kampą θ , reiškiantį segmento sulenkimo kampą apie y ašį, toliau pagal tai buvo sudaromi skaičiavimai ir parenkamos formulės.

Nurodžius pirmojo taško (užduoties taškas), kurį reikia pasiekti 3-jam straublio tipo roboto segmentui koordinatės (x_g, y_g, z_g) ir iš jo einančio vienetinio vektoriaus (krypties vektorius) taško koordinatės (x_{g1}, y_{g1}, z_{g1}) pirmaisiai yra surandami kampai: užduoties taško kampas apie z ašį – ϕ_1 , krypties vektoriaus pasukimo kampas apie y ašį – α , krypties vektoriaus pasukimo kampas apie z ašį – β (žr. 23 pav.). Šie kampai ieškomi iš stačiakampių trikampių, nubraižytų pagal roboto pirmo segmento pradžios taško O $(0, 0, 0)$, užduoties ir krypties taškų turimas koordinatės, skaičiuojant tangentes. Rašant programos kodą šiems kampams rasti, taip pat buvo nustatoma kuriems ketvirčiams priklauso užduoties ir krypties vektoriaus taškai, tam, kad būtų galima užrašyti iš tangentių gautus kampus ϕ_1 , α ir β ribose nuo 0° iki 360° . Reikėjo nustatyti, kuriam ketvirčiui taško O atžvilgiu XY plokštumoje priklauso užduoties taškas ir kuriems ketvirčiams užduoties taško atžvilgiu priklauso krypties vektoriaus taškas XY ir XZ plokštumose. Visa tai įvertinta pagal užduoties ir krypties vektorių taškų x, y ir z koordinatinių teigiamus arba neigiamus ženklus.



23 pav. Užduoties taškas, krypties vektorius ir kampai ϕ_1 , α ir β Dekarto koordinačių plokštumoje, a) vaizdas iš viršaus, b) vaizdas iš šono

Toliau užduoties taškas yra pasukamas į XZ plokštumą. Ieškant straublio tipo roboto segmento kampų ϕ , θ buvo nuspręsta taikyti $R_{x,\alpha}$, $R_{y,\alpha}$, $R_{z,\alpha}$ posūkio matricas (13)(14)(15).[22]

$$R_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (13)$$

$$R_{y,\alpha} = \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & 1 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \quad (14)$$

$$R_{z,\alpha} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (15)$$

Bet prieš pradėdant taikyti $R_{x,\alpha}$, $R_{y,\alpha}$, $R_{z,\alpha}$ posūkio matricas (13)(14)(15), pirmiausia uždavinys buvo išspręstas dvimatėje XY plokštumoje, siekiant suplanuoti skaičiavimų eigą ir vizualiai robotą pamatyti dvimatėje plokštumoje.

2.4. Straublio tipo roboto 3-jų segmentų išsilenkimo kampų paieškos uždavinys dvimatėje plokštumoje

Taikant programą „Wolfram Mathematica“ 11.2 versiją, buvo sprendžiamas 3-jų segmentų straublio tipo roboto segmentų sulenkimo kampų paieška dvimatėje plokštumoje, kai yra nurodytas užduoties taško, krypties vektoriaus taško koordinatės ir apibrėžti pirmo, antro ir trečio segmentų ilgiai, kurie visi yra to paties ilgio ir lygūs L . Čia naudojama tik viena posūkio matrica $R(\gamma)$ (16), kuri tašką dvimatėje plokštumoje suka apie koordinatinių plokštumos tašką $(0,0)$ prieš laikrodžio rodyklę.

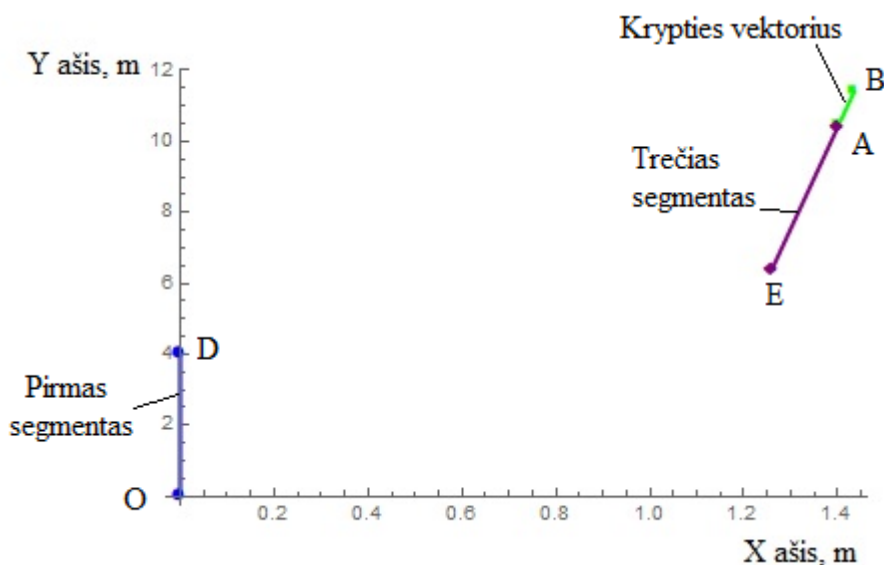
$$R(\gamma) = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) \\ \sin(\gamma) & \cos(\gamma) \end{bmatrix} \quad (16)$$

Iš žinomų krypties taško B vektoriaus koordinatinių (x_{g1}, y_{g1}) atimamos užduoties taško A koordinatės (x_g, y_g) ir gaunamas krypties vektorius, einantis iš taško $(0,0)$ į tašką C. Pritaikius stačiakampio trikampio tangento formulę, gaunamas kampas ψ tarp krypties vektoriaus taško C ir x ašies. Gaunami pirmas ir trečias segmentai savo pradinėse padėtyse (kai jie yra išsitiesę). Pirmo segmento pradinės padėties taškas – D, gaunamas tašką $(L, 0)$ pasukus su posūkio matrica $R(90^\circ)$ į y ašį (17). Trečio segmento pradinės padėties taškas E yra gaunamas tašką $(L, 0)$ pasukus $R(\gamma_1 + 180^\circ)$ ir pridėjus prie taško A koordinatinių (18)(19). Gautas pradinis pirmo ir trečio segmentų vaizdas yra atvaizduojamas „Wolfram Mathematica“ programos lange (žr. 24 pav.).

$$D = (L, 0) \cdot R(90^\circ) \quad (17)$$

$$E' = ((L, 0) \cdot R(\psi + 180^\circ)) + A \quad (18)$$

$$E = E' + A \quad (19)$$



24 pav. Straublio tipo roboto 1 ir 3 segmentų atvaizdavimas pradinėse padėtyse dvimatėje plokštumoje

Iš 24 paveikslėlio matyti, jog tiek pirmas, tiek trečias segmentai gali lenktis į dvi puses, pavyzdžiui, pirmas segmentas gali lenktis į dešinę arba į kairę pusę nuo y ašies. Todėl į tai atsižvelgiant bus

ieškomi 4 segmentų išsilenkimo variantai, tai yra, apskritimo lankai. Parinkus pirmo segmento sulenkimo kampą γ_1 , o trečio segmento sulenkimo kampą γ_3 , yra surandami apskritimo lankų, t.y. išlenktų segmentų, spinduliai R_1 (1-o segmento) ir R_2 (2-o segmento) su formule (20).

$$R_{ji} = \frac{L_j}{\gamma_j} \quad (20)$$

čia R_{ji} – yra j-ojo segmento i-tasis spindulys, γ_j – j-ojo segmento apskritimo lanko kampas, L – j-ojo segmento stuburo ilgis.

Tada, pirmo ir trečio segmento spinduliai yra pasukami $+90^\circ$ ir -90° kampu į abi puses nuo jų pradinių padėčių su formule (20). Reikia turėti omenyje, jog pasukus trečio segmento spindulį pagal formulę (21) gautą spindulio tašką pridėti prie užduoties taško A koordinatėms.

$$W_{ji} = (R_{ji} \cdot G) \cdot \mathbf{R}(\pm 90^\circ) \quad (21)$$

čia W_{ji} yra j-ojo segmento pasuktas i-tasis spindulys, G – pradinio pirmo arba trečio segmento vienetinių vektorių taškų koordinatės, kai vektoriai išeina iš (0,0) taško (D arba E' vektorių taškų koordinatės padalintos iš L). Toliau reikia pasukti segmentų apskritimo lankų spindulius apie gautus pirmo ir trečio spindulių taškus iš (21) formulės. Pirmo segmento spinduliai pasukami vienodu kampu γ_2 , o trečiojo segmento spinduliai pasukami kampu γ_3 taikant šias dvi formules (22)(23). Reikia turėti omenyje, jog trečio segmento atveju taikant (22) ir (23) formules posūkių matricose reikia papildomai prie šių kampų pridėti kampą lygų $\psi + 90^\circ$ (24)(25), be to, trečio segmento atveju (22)(23) skaičiavimo formulės sukeičiamos vietomis. Pagal gautas pirmo ir trečio 4 atvejų spindulių koordinates yra nubraižomi pirmo ir trečio segmentų 4 išsilankstymai (žr. 25 pav.).

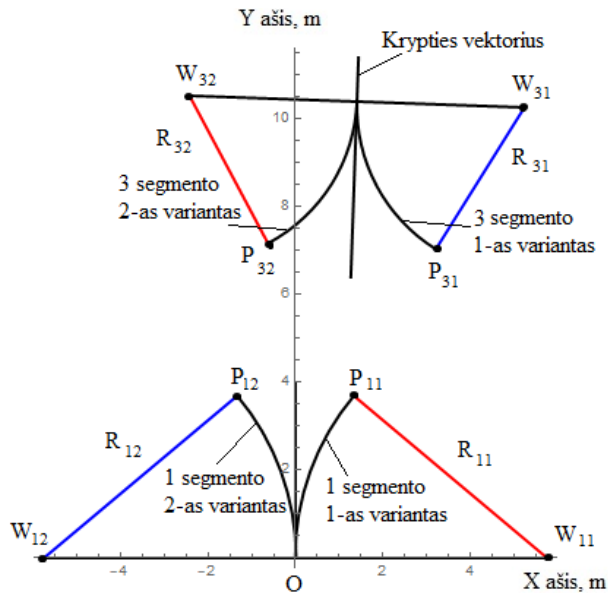
$$P_{11} = (R_j \cdot (1,0)) \cdot \mathbf{R}(180^\circ - \gamma_j) + W_{j1} \quad (22)$$

$$P_{12} = (R_j \cdot (1,0)) \cdot \mathbf{R}(\gamma_j) + W_{j2} \quad (23)$$

$$P_{31} = (R_j \cdot (1,0)) \cdot \mathbf{R}(\gamma_j + \psi + 90^\circ) + W_{j2} \quad (24)$$

$$P_{32} = (R_j \cdot (1,0)) \cdot \mathbf{R}((180^\circ - \gamma_j) + \psi + 90^\circ) + W_{j1} \quad (25)$$

čia P_{11} ir P_{12} – 1-o segmento 1-o ir 2-o spindulių, pasuktų kampu γ_j apie W_{j1} ir W_{j2} taškus, taškų koordinatės, P_{31} ir P_{32} – 3-o segmento 1-o ir 2-o spindulių, pasuktų kampu γ_j apie W_{j1} ir W_{j2} taškus, taškų koordinatės.



25 pav. Straublio tipo roboto 1 ir 3 segmentų 4-ių išsilenkimo variantų atvaizdavimas dvimatėje plokštumoje
Toliau gaunamos pirmo ir trečio segmentų 4-ių variantų apskritimo lankų liestinės (žr. 26 pav.), išreiškiančios galimų segmentų išsilankstymo variantų kryptis. Šios liestinės, šiuo atveju, skaičiuojamos kaip vienetiniai vektoriai pritaikant formules (26)(27)(28)(29). Kadangi pastarosios formulės gauna liestinių vektorius einančius iš taško (0,0), todėl siekiant vizualiai gauti jų padėti XY plokštumoje prie gaunamų taškų reikia pridėti atitinkamus P_{ji} taškus gautus iš formulių (22)(23)(24)(25).

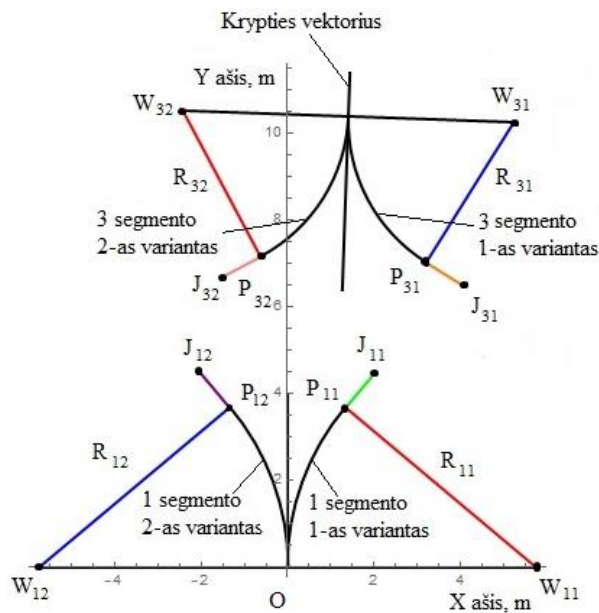
$$J_{11} = (1,0) \cdot R(180^\circ - \gamma_j - 90^\circ) \quad (26)$$

$$J_{12} = (1,0) \cdot R(\gamma_j + 90^\circ) \quad (27)$$

$$J_{31} = (1,0) \cdot R(\gamma_j + \psi + 180^\circ) \quad (28)$$

$$J_{32} = (1,0) \cdot R((180^\circ - \gamma_j) + \psi) \quad (29)$$

čia J_{11} ir J_{12} – 1-o segmento 1-o ir 2-o variantų apskritimo lankų liestinės išreikštos vektoriais, J_{31} ir J_{32} – 3-o segmento 1-o ir 2-o apskritimo lankų liestinės išreikštos vektoriais



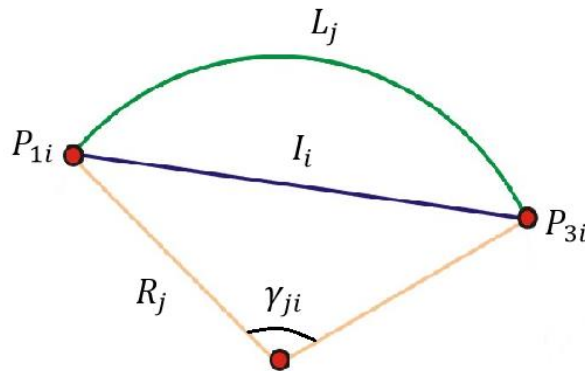
26 pav. Straublio tipo roboto 1 ir 3 segmentų 4-ių išsilenkimo variantų su jų kryptis nurodančiais vektoriais atvaizdavimas dvimatėje plokštumoje

Dabar ieškomi atkarpų, esančių tarp pirmo ir trečio segmentų galimų išsilankstymo variantų P_{ji} taškų, ilgiai. Kaip matoma iš 26 paveikslėlio, galimi 4 atkarpų variantai tarp pirmo ir trečio segmentų: 1) atkarpa tarp taškų P_{11} ir P_{31} , 2) atkarpa tarp taškų P_{11} ir P_{32} , 3) atkarpa tarp taškų P_{12} ir P_{31} , 4) atkarpa tarp taškų P_{12} ir P_{32} . Šių atkarpų ilgiai surandami atėmus vieno taško koordinates iš kito taško koordinatėms ir pritaikius vektoriaus ilgio skaičiavimo formulę, gaunama (30) formulė.

$$I_i = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (30)$$

čia I_i – i -osios atkarpos ilgis, x_1, y_1 – atkarpos pirmo taško koordinatės, x_2, y_2 – atkarpos antro taško koordinatės.

Toliau, reikia surasti tokį antro segmento apskritimo lanko kampą γ_{ji} tarp jo spindulių R_j , kuris sujungtų pirmo ir trečio segmentų taškus P_{1i} ir P_{3i} taip, kad antro segmento apskritimo lanko ilgis būtų lygus iš anksto apibrėžtam segmento stuburo ilgiui L_j (žr. 27 pav.). Ieškant kampo γ_{ji} yra pasitelkiama apskritimo lanko stygos skaičiavimo formulė (31). Kadangi pagal (31) formulę yra du nežinomieji γ_{ji} ir R_j , nežinomųjų skaičius R_j sumažinamas, išreiškiant per (11) formulę ir gaunama išraiška (32). Dabar yra likęs tik vienas nežinomasis γ_{ji} , tačiau, jo išsireikšti iš gautos formulės negalima.



27 pav. Roboto antrą segmentą, esančio apskritimo lanko formos, apibūdinantys parametrai

$$I_i = 2R_j \sin\left(\frac{\gamma_{ji}}{2}\right) \quad (31)$$

$$I_i = 2 \frac{L_j}{\gamma_{ji}} \sin\left(\frac{\gamma_{ji}}{2}\right) \quad (32)$$

Literatūros šaltinyje [29], šiai problemai spręsti yra daugiausiai kalbama apie Niutono metodą. Niutono metodas yra artėjimo metodas, kurį taikant yra ieškoma funkcijos $f(x)$ nario x vertė, kai $f(x) = 0$. Kad būtų galima pritaikyti šį metodą (32) formulės nežinomam kampui γ_{ji} rasti, formulė išreiškiama kaip funkcija (33), o kampas γ_{ji} dėl patogumo pažymimas kaip x . Po to funkciją perdarius į (34) yra surandama funkcijos išvestinė (35). Tada taikoma Niutono formulė (36), tačiau jai reikia parinkti kokį nors pradinį tašką iš intervalo $[0, 2\pi]$ ir atliekama tiek iteracijų (pakartojimų), kol randamas toks x , prie kurio $f(x)$ yra arčiausiai arba lygus 0. Tačiau Niutono metodas turi trūkumą - blogai parinkus pradinį tašką, gali tekti atlikti labai daug iteracijų arba metodas diverguoja, tai yra, nėra surandamas teisingas atsakymas. Todėl tinkamam pradiniam taškui parinkti yra siūloma paimti vidurinę x kitimo intervalo reikšmę, gauti funkcijos $f(x)$ grafiką ir iš ten paimti x reikšmę arba atlikti pirmą aproksimaciją kitu paieškos metodu ir su Niutono metodu pabaigti ieškoti atsakymo.

$$f(x) = \frac{1}{x} \sin\left(\frac{x}{2}\right) - \frac{I_i}{2L_j} \quad (33)$$

$$f(x) = 1 - \frac{2L_j}{xI_i} \sin\left(\frac{x}{2}\right) \quad (34)$$

$$f'(x) = \frac{2L_j \sin\left(\frac{x}{2}\right)}{x^2 I_i} - \frac{L_j \cos\left(\frac{x}{2}\right)}{x I_i} \quad (35)$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (36)$$

čia x yra kampas γ_{ji} radianais.

Buvo pabandyti keli atvejai Niutono metodo pradiniam taškui gauti, vienas atvejis kai nurodomas x intervalo vidurio taškas, o kitas kai pradinis taškas surandamas pritaikius nuoseklų paieškos metodą. Tačiau atlikus daugiau bandymų pasirodė, jog Niutono metodas yra gana nepatikimas ir lėtas šiuo atveju ieškomam kampui γ_{ji} rasti, todėl buvo išbandytas kitoks (32) formulės sprendimo metodas, kurio randamas kampo γ_{ji} atsakymas yra mažesnio tikslumo, tačiau šiuo atveju patikimesnis, nes visuomet yra randamas atsakymas. Tai yra sprendimas išskleidus $f(x)$ funkciją (33) Teiloro eilute.[30]

Pirmiausia, $\sin\left(\frac{x}{2}\right)$ išskleidžiam Teiloro eilute (37), iš jos paimami pirmi trys nariai ir jie padalinami iš x (38). Tada, atsižvelgiant į (33) formulę, lygtis užrašoma taip (39). Visus lygties narius sukėlus į vieną pusę atliekamas $a = x^2$ pakeitimas ir gaunama tokia lygtis (40). Išsprendus kvadratinę lygtį yra pasirenkamas mažesnis gautas sprendinys ir iš jo ištraukus kvadratinę šaknį gaunamas atsakymas su nedidele paklauda.

$$\sin\left(\frac{x}{2}\right) = \frac{1}{2}x - \frac{1}{48}x^3 + \frac{1}{3840}x^5 - \frac{1}{3645120}x^7 + \dots \quad (37)$$

$$\frac{\sin\left(\frac{x}{2}\right)}{x} = \frac{1}{2} - \frac{1}{48}x^2 + \frac{1}{3840}x^4 \quad (38)$$

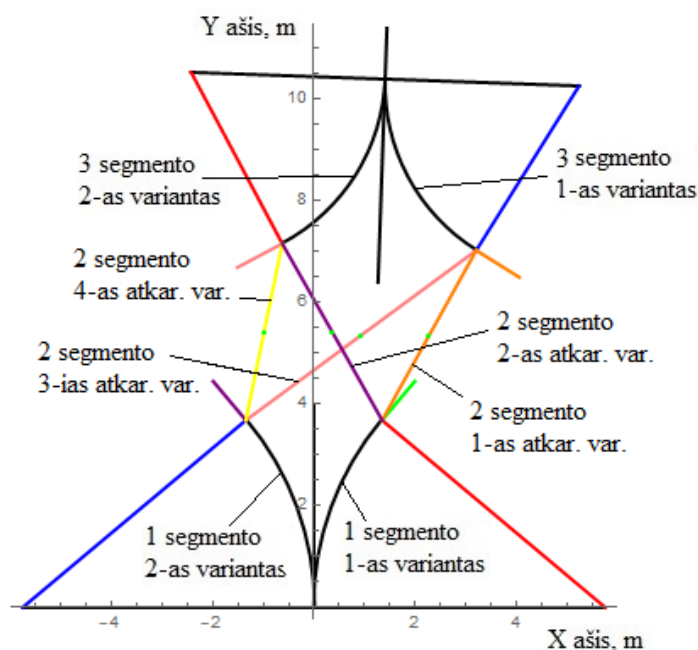
$$\frac{1}{2} - \frac{1}{48}x^2 + \frac{1}{3840}x^4 = C \quad (39)$$

čia C yra lygu $\frac{I_i}{2L_j}$.

$$\frac{1}{3840}a^2 + \frac{1}{48}a + \left(\frac{1}{2} - C\right) = 0 \quad (40)$$

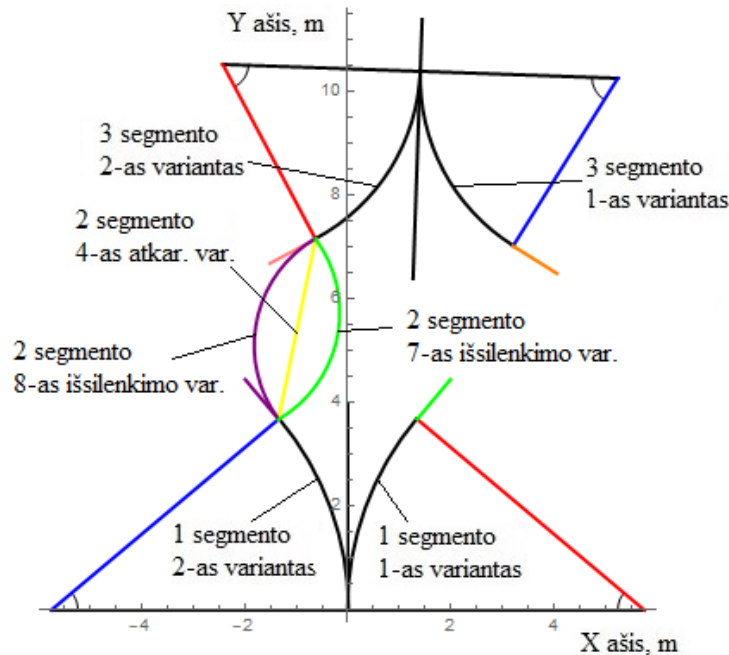
čia a yra lygu x^2 .

Skaičiuojant antro roboto segmento išsilenkimo kampą γ_{ji} kuriuo nors iš prieš tai aprašytu metodu, pastebėta, jog kuriant programos algoritmą, skirtą straublio tipo robotui valdyti, būtinai reikia nurodyti sąlygą, jog skaičiuoti šiuos antro segmento kampus galima tik tada, kai rastas atkarpos I_i yra trumpesnis už iš anksto apibrėžtą antro segmento stuburo ilgį L_j . Taip yra sumažinamas antro segmento galimų variantų skaičius ir nedaromi negalimi antro segmento kampų paieškos skaičiavimai. Toliau randami antro segmento apskritimo lanko spinduliai iš (20) formulės. Prieš tęsiant skaičiavimus, yra apžvelgiami antro segmento atkarpų, jungiančių pirmo ir trečio išlenktų segmentų galų taškus, išsidėstymai dvimatėje XY plokštumoje (žr. 28 pav.).



28 pav. Straublio tipo roboto 1 ir 3 segmentų 4-ių išsilenkimo variantų su jų kryptis nurodančiais vektoriais bei juos jungiančių atkarpų 4 variantų atvaizdavimas dvimatėje plokštumoje

Iš 28 paveikslėlio matyti, jog kiekviena atkarpa, jungianti tam tikrą pirmą arba trečią segmento išsilenkimo variantą, gali turėti po du antro segmento išsilenkimo variantus. Tai yra, kiekvienai antro segmento atkarpai iš kairės ir iš dešinės pusės yra po vieną, vienas kitam išsilenkimo atžvilgiu identišką, antro segmento išsilenkimo variantą (žr. 29 pav.). Antro segmento vizualizavimui realizuoti ir tolimesniems skaičiavimams reikia surasti antro segmento 4-ių atkarpų kampus χ_i (χ_i – indeksas i žymi antro segmento atkarpos varianto numerį) apie tašką (0,0). Kampai χ_i randami tokiu pačiu principu, kaip buvo ieškomas krypties vektoriaus kampas ψ - pirmiausia atimant iš atitinkamo taško koordinatę P_{3i} taško koordinatę P_{1i} , taip gaunant atitinkamo antro segmento atkarpos varianto vektorių einantį iš taško (0,0). Tada kampas χ_i paskaičiuojamas pritaikant stačiakampio trikampio tangento skaičiavimo formulę ir nustatius, kuriame XY koordinatinių plokštumos ketvirtyje yra antro segmento atkarpos vektoriaus taškas prie gauto kampo atitinkamai pridėdamas vienas iš šių kampų: $0^\circ, 90^\circ, 180^\circ, 270^\circ$.



29 pav. Straublio tipo roboto 1 ir 3 segmentų 4-ių išsilenkimo variantų su jų kryptis nurodančiais vektoriais bei antro segmento 4 varianto dviejų segmento stuburo išsilenkimo variantų atvaizdavimas dvimatėje plokštumoje

Kad būtų galima pavaizduoti kiekvieną galimą antro segmento išsilenkimo variantą (iš viso matematiškai jų yra galimi 8), kiekvienam variantui reikia rasti bent po vieno apskritimo spindulio taško koordinatas. Tai padaroma skaičiavimui pritaikius vieną iš šių formulių (22)(23)(24)(25), vietoje W_{j1} arba W_{j2} taškų atitinkamą P_{1i} tašką ir posūkio matricoje \mathbf{R} įstačius kiekvienam variantui atitinkamus kampus (čia visi kampai išreikšti laipsniais): $\chi_1 + (90^\circ - (\frac{\gamma_{21}}{2}))$, $\chi_1 - (90^\circ - (\frac{\gamma_{21}}{2}))$, $\chi_2 + (90^\circ - (\frac{\gamma_{22}}{2}))$, $\chi_2 - (90^\circ - (\frac{\gamma_{22}}{2}))$, $\chi_3 + (90^\circ - (\frac{\gamma_{23}}{2}))$, $\chi_3 - (90^\circ - (\frac{\gamma_{23}}{2}))$, $\chi_4 + (90^\circ - (\frac{\gamma_{24}}{2}))$, $\chi_4 - (90^\circ - (\frac{\gamma_{24}}{2}))$.

Paskutinė uždavinio dalis ieškant trijų segmentų išsilenkimo kampų yra antro segmento surastų galimų išsilenkimo variantų, segmentų pradžios ir galo orientacijų suradimas išreikšus jas vienetiniais vektoriais ir šių gautų vektorių palyginimas su atitinkamais pirmo ir trečio segmento išsilenkimo variantų kryptis išreiškiančiais vienetiniais vektoriais.

Antro segmento kiekvieno varianto krypties vektoriai gaunami koreguojant (26) formulės posūkio matricų kampus, gaunant štai tokias apibendrintas formules (41)(42)(43)(44). Visiems antro segmento stuburo išsilenkimo variantams, kurių numeris yra nelyginis skaičius, taikomos (41)(42) formulės, o antro segmento išsilenkimo variantams, kurių numeriai yra lyginiai skaičiai, taikomos (43)(44) formulės. Kai norima atvaizduoti šiuos gautus vienetinius krypties vektorius grafiškai, reikia nepamiršti juos pridėti prie atitinkamų P_{1i} arba P_{3i} taškų koordinatų. Atvaizduojami du antro segmento išsilenkimo variantai, kurie grafiškai žiūrint sujungia pirmo ir trečio segmento išsilenkimo variantus taip, jog segmentų kryptis išreiškiantys vienetiniai vektoriai esantys tarp pirmo ir antro segmentų bei tarp antro ir trečio segmentų tarpusavyje yra nukreipti pilnai arba beveik (su didelėmis paklaidomis) priešingomis kryptimis (žr. 30 pav.).

$$K_{e1} = (1,0) \cdot \mathbf{R} \left(\chi_{e+1} + \left(90^\circ - \left(\frac{\gamma_{e+1}}{2} \right) \right) + 90^\circ \right) \quad (41)$$

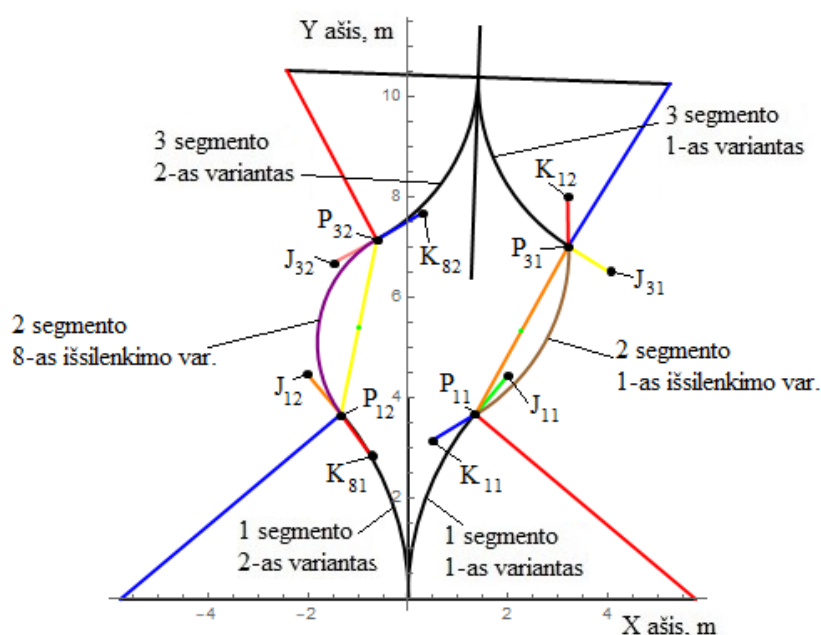
$$K_{e2} = (1,0) \cdot \mathbf{R} \left(\chi_{\frac{e+1}{2}} + \left(\frac{\gamma_{e+1}}{2} \right) \right) \quad (42)$$

čia K_{e1} ir K_{e2} – yra antro segmento e-ojo išsilenkimo (kai e nelyginis skaičius) pradžios ir galo kryptių vienetiniai vektoriai, $\chi_{\frac{e+1}{2}}$ – antrojo segmento e-ojo išsilenkimo varianto kampas tarp jo atkarpos, jungiančios 1 ir 3 segmentų išsilenkimo variantus, vektoriaus ir +x ašies (kai e nelyginis skaičius), $\gamma_{\frac{e+1}{2}}$ – antro segmento e-ojo išsilenkimo varianto segmento stuburo sulenkimo kampas (kai e nelyginis skaičius).

$$K_{e1} = (1,0) \cdot \mathbf{R} \left(\chi_{\frac{e}{2}} - \left(90^\circ - \left(\frac{\gamma_e}{2} \right) \right) - 90^\circ \right) \quad (43)$$

$$K_{e2} = (1,0) \cdot \mathbf{R} \left(\chi_{\frac{e}{2}} + \left(\frac{\gamma_e}{2} \right) \right) \quad (44)$$

čia K_{e1} ir K_{e2} – yra antro segmento e-ojo išsilenkimo (kai e lyginis skaičius) pradžios ir galo kryptių vienetiniai vektoriai, $\chi_{\frac{e}{2}}$ – antrojo segmento e-ojo išsilenkimo varianto kampas tarp jo atkarpos, jungiančios 1 ir 3 segmentų išsilenkimo variantus, vektoriaus ir +x ašies (kai e lyginis skaičius), $\gamma_{\frac{e}{2}}$ – antro segmento e-ojo išsilenkimo varianto segmento stuburo sulenkimo kampas (kai e lyginis skaičius).



30 pav. Straublio tipo roboto 1 ir 3 segmentų 4-ių išsilenkimo variantų su jų kryptis nurodančiais vektoriais bei antro segmento 2-jų išsilenkimo variantų su jų kryptis nurodančiais vektoriais atvaizdavimas dvimatėje plokštumoje

Paskutinėje uždavinio dalyje palyginama kiekvieno gauto antro segmento išsilenkimo varianto galų kryptių vienetinių vektorių kryptis su atitinkamais pirmo ir trečio segmentų galų kryptių vienetinių vektorių kryptimis. Pavyzdžiui, antro segmento 8 varianto atveju (žr. 30 pav.), pirmo segmento antro varianto galo krypties vienetinio vektoriaus J_{12} kryptis palyginama su antro segmento 8-o išsilenkimo varianto 1-uoju vienetinio krypties vektoriaus K_{81} kryptimi, o trečio segmento antro varianto vienetinio vektoriaus J_{32} kryptį palyginama su antro segmento 8-o išsilenkimo varianto 2-uoju vienetinio vektoriaus K_{82} kryptimi. Palyginimui taikoma vektorinę sandaugą (angl. *cross product*)

(45). Kadangi dvimatėje plokštumoje taškai turi tik dvi koordinates x ir y , tai z koordinatė paliekama lygi 0, todėl iš vektorinės sandaugos, šiuo atveju, gaunamas atsakymas, lygus $(0, 0, N)$, kur N – realusis skaičius. Vektorinė sandauga atsakymą pateikia vektoriaus koordinatėmis \vec{c} , todėl siekiant išreikšti atsakymą vienu realiu skaičiumi, yra suskaičiuojamas gauto \vec{c} vektoriaus ilgis, taikant vektoriaus ilgio skaičiavimo formulę (46).

$$\vec{c} = \vec{a} \times \vec{b} = (a_y b_z - a_z b_y, -(a_x b_z - a_z b_x), a_x b_y - a_y b_x) \quad (45)$$

$$l = \sqrt{x^2 + y^2 + z^2} \quad (46)$$

čia \vec{c} yra vektorinės sandaugos apskaičiuotas \vec{c} vektorius, \vec{a} ir \vec{b} yra pirmas ir antras vektoriai, l yra vektoriaus ilgis.

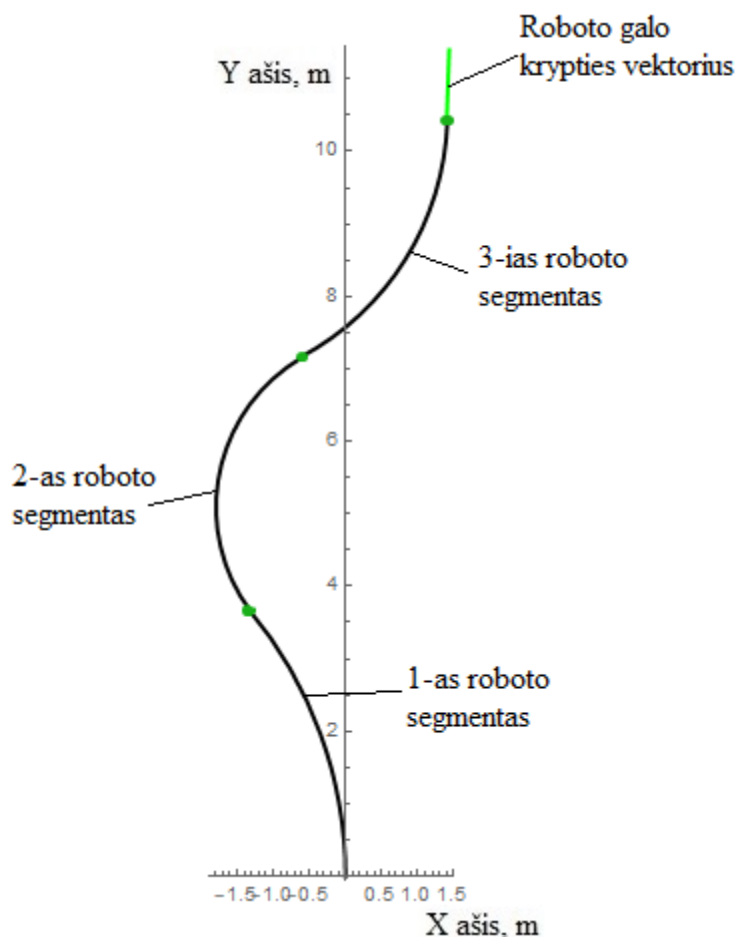
Gaunant atsakymus iš lyginamų vienetinių, segmentų kryptis nusakančių, vektorių vektorinių sandaugų ir apskaičiuotus gautų atsakymų vektorių ilgius, siekiant įvertinti, ar buvo surasti geri visų trijų segmentų išsilankstymo kampai, tai yra, ar visi trys segmentai kuriame nors variante susijungia tvarkingai į nuoseklią roboto grandinę, yra nusprendžiama pagal vektorinei sandaugai būdingą savybę. Ši vektorinės sandaugos savybė nurodo, jog jeigu du vektoriai \vec{a} ir \vec{b} yra kolinearūs, tai yra, vienas kitam lygiagretūs ($\vec{a} \parallel \vec{b}$), tada vektorinė sandauga yra lygi nuliui ($\vec{a} \times \vec{b} = 0$). Kai vienetiniai vektoriai \vec{a} ir \vec{b} yra vienas kitam statmeni ($\vec{a} \perp \vec{b}$), apskaičiuotus jų vektorinę sandaugą ir po to paskaičiuotus vektoriaus ilgį l visada gaunamas 1. Ribose $[0, 1]$ visada bus gaunamas vektoriaus \vec{c} ilgis, kol \vec{a} ir \vec{b} vektorinės sandaugos vektoriai bus vienetiniai, todėl šiomis vektoriaus ilgio ribomis išreiškiama dviejų vektorių kolinearumo paklaida nuo 0 % iki 100 %.

Paskutinė problema, kuri yra susijusi su tinkamų antro segmento išsilankstymo variantų parinkimu, yra matoma 29 paveikslėlyje. Antras segmentas turi du vienodų kampų išsilankstymo variantus ant vienos antro segmento atkarpos, tai yra, 7 ir 8 variantus. Antro segmento 7-o išsilankstymo varianto vektorinių sandaugų vektorių ilgiai yra didesni už antro segmento 8-o varianto už iš vektorinių sandaugų gautus vektorių ilgius, todėl 7-as išsilankstymo variantas vien dėl to yra atmetamas. Tačiau, jeigu ieškomi roboto segmentų išsilankstymo kampai prie kitų užduoties taškų ir krypties vektoriaus koordinačių, vėl įvyksta panaši situacija, kaip yra pademonstruota 29 paveikslėlyje, tačiau šįkart antro segmento 7-o varianto vektorinių sandaugų gautų vektorių ilgiai yra mažesni už 8-o išsilankstymo varianto gautus vektorinių sandaugų atsakymus. Grafiškai žiūrint, toks roboto variantas būtų neįmanomas, nes pirmo segmento 2-o varianto vienetinio vektoriaus kryptis sutaptų su antro segmento 7-o varianto pirmojo vienetinio vektoriaus kryptimi, tačiau pagal vektorinę sandaugą šis roboto išsilankstymo variantas būtų galimas. Tam, kad būtų galima įvertinti matematiškai, ar surastas roboto trijų išsilankstusių segmentų yra grafiškai įmanomas išlankstyti, tai yra, ar surasto roboto varianto pirmo ir antro segmentų bei antro ir trečio segmentų vienetiniai kryptinių vektoriai žiūri į priešingas puses, yra pasitelkiama skaliarinė vektorių sandauga (angl. *dot product*). Taikant skaliarinės sandaugos formulę (47) yra apskaičiuojamas kampas tarp dviejų vektorių. Jeigu vektoriai žiūri į tą pačią pusę, gautas kampas ϑ radianais yra teigiamas, jeigu vektoriai yra statmeni vienas kitam $\vartheta = 0$, o jeigu vektoriai žiūri į priešingas puses, kampas ϑ yra gaunamas neigiamas. Remiantis gauto kampo ϑ ženklu yra nustatoma, ar grafiškai žiūrint surastas roboto išlenktų segmentų išsidėstymo variantas yra galimas, tai yra, ar grafiškai segmentai gali susijungti, net jeigu vektorių sandaugų apskaičiuotų vektorių ilgiai yra lygūs 0.

$$\vec{a} \cdot \vec{b} = x_a x_b + y_a y_b + z_a z_b \quad (47)$$

čia \vec{a} ir \vec{b} yra pirmas ir antras vektoriai, x_n, y_n, z_n yra n-ojo vektoriaus koordinatės.

Įvertinus pirmo ir antro segmentų vienetinių krypčių vektorių bei antro ir trečio vienetinių krypčių vektorių vektorinių ir skaliarinių sandaugų rezultatus iš 30 paveikslėlio parenkamas toks roboto variantas, kai jį sudaro pirmo segmento 2-as variantas, antro segmento 8-as variantas, trečio segmento 2-as variantas (žr. 31 pav.) ir šių segmentų išsilenkimo kampai yra uždavinio atsakymas.



31 pav. Išrinktas galutinis roboto 3-jų segmentų variantas pagal iš anksto nurodytas užduoties taško koordinates ir roboto galo krypties vektoriaus koordinates dvimatėje XY plokštumoje

Poskyrio išvados:

- Išsprendus roboto straublio trijų segmentų išsilenkimo kampų paieškos uždavinį dvimatėje plokštumoje, nustatyta skaičiavimo eiga, kurios laikantis turi būti kuriamas straublio tipo roboto 3-jų segmentų valdymo programos algoritmas.
- Nustatyta, jog gali būti 2 pirmo segmento išsilenkimo variantai, 2 trečio segmento išsilenkimo variantai ir 8 antro segmento išsilenkimo variantai. Kai užduoties koordinatės yra visada nurodomos viename ketvirtyje tarp +x ir +y ašies, antro segmento 5-as ir 7-as variantai yra visada sprendžiant iš to, jog antro segmento 5-o ir 7-o variantų galų krypties vienetiniai vektoriai nukreipti ta pačia kryptimi, kaip atitinkamų pirmo ir trečio segmento variantų galų vienetinių vektorių kryptys. Todėl pagal tai nuspręsta kuriant straublio tipo roboto valdymo algoritmą užduoties koordinates visada atsukti į tą patį vieną koordinačių plokštumą, kad būtų galima sumažinti antro segmento galimų išsilankstymo variantų kiekį iki 6 variantų.

- Prieš pradėdant ieškoti antro segmento kiekvieno išsilenkimo varianto išsilenkimo kampo reikia patikrinti sąlygą, ar atstumas tarp atitinkamo pirmo segmento išsilenkimo varianto ir trečio segmento išsilenkimo varianto yra mažesnis už iš anksto nurodytą antro segmento stuburo ilgį. Tai, priklausomai nuo užduoties koordinačių, gali sumažinti arba palikti tokį patį antro segmento išsilenkimo variantų skaičių (pavyzdžiui, užduoties koordinatėms esant toliau roboto bazės centro taško, gali likti 4 arba tik 2 antro segmento išsilenkimo skaičiavimo variantai).
- Nustatyta, jog antro segmento kampą galima rasti iš (33) formulės funkcijos, sprendžiant funkciją vienu iš dviejų metodų: Niutono metodu arba išskleidus funkciją Teiloro eilute ir paėmus pirmus tris šios eilutės narius.
- Parinkti tinkamiausiam roboto variantui iš esamų 2-jų pirmo segmento išsilankstymo variantų, 6-ių antro segmento išsilankstymo variantų ir 2-ų trečio segmento išsilankstymo variantų, reikia taikyti vektorių vektorinę ir skaliarinę sandaugą. Vektorinės sandaugos atveju taikoma vektorinės sandaugos dviejų vektorių kolinearumo savybė, kad gautieji vektorinių sandaugų vektoriai būtų kuo artimesni arba lygūs 0, o skaliarinės sandaugos atveju žiūrima, jog gautas kampas radianais būtinai būtų neigiamas.

2.5. Straublio tipo roboto 3-jų segmentų išsilenkimo ir pasukimo kampų paieškos uždavinys trimatėje aplinkoje ir roboto valdymo algoritmo kūrimas

Išsprendus straublio tipo roboto 3-jų segmentų išsilenkimo kampų paieškos uždavinį dvimatėje XY plokštumoje, toliau pagal tai sudarinėjama skaičiavimo eiga trimatėje Dekarto koordinačių sistemoje, kuri buvo pradėta kurti 2.3 poskyryje. Pastarajame poskyryje taip pat buvo pirmiausiai surasti tokie kampai kaip ϕ_1 – užduoties taško kampas apie z ašį, α – krypties vektoriaus pasukimo kampas apie y ašį, β – krypties vektoriaus pasukimo kampas apie z ašį. Toliau reikėjo pasukti užduoties tašką, kurio koordinatės (xg, yg, zg), į XZ plokštumą su surastu kampu ϕ_1 , tačiau 2.3 poskyryje ties šia vieta uždavinys buvo nutrauktas. Iš 2.4 poskyryje pateiktų išvadų matome, jog atsukus užduoties tašką visada į tą patį vieną koordinačių plokštumos ketvirtį galima sumažinti skaičiavimo skaičių, nes sumažinamas antro segmento variantų skaičius iš 8 į 6. Taip pat yra dar viena priežastis, tai palengvina supratimą, kokias posūkio matricas kiekvienam iš trijų roboto segmentui reikia taikyti sprendžiant uždavinį.

Dvimatėje plokštumoje visi vektorių taškai yra išdėstyti vienoje plokštumoje ir posūkio matrica $\mathbf{R}(\gamma)$, (16) formulė, taškus sukdamo tik apie vieną tašką (0,0) (menamoji z ašis). Trimatėje plokštumoje yra trys posūkio matricos $\mathbf{R}_{x,a}$, $\mathbf{R}_{y,a}$, $\mathbf{R}_{z,a}$ apie kiekvieną ašį (13)(14)(15). Taikant bandymų ir klaidų metodą ir sukiojant vektorių, kurio taško koordinatės yra lygios vieno segmento einančio iš roboto bazės taško koordinatėms (pavyzdžiui žr. 18 pav. išlenkto segmento galo taškas), su posūkio matricomis $\mathbf{R}_{x,a}$, $\mathbf{R}_{y,a}$, $\mathbf{R}_{z,a}$, buvo prieita prie tokių posūkio matricų panaudojimų variantų kiekvienam iš trijų atskirai roboto segmentui išlenkti arba pasukti. Pirmą segmentą reikia lenkti su $\mathbf{R}_{y,a}$, sukuti su $\mathbf{R}_{z,a}$ posūkio matricomis, antrą segmentą lenkti su $\mathbf{R}_{y,a}$, sukuti su $\mathbf{R}_{x,a}$ posūkio matricomis ir trečią segmentą lenkti su $\mathbf{R}_{y,a}$, o sukuti su $\mathbf{R}_{y,a}$, $\mathbf{R}_{x,\alpha}$ ir $\mathbf{R}_{z,a}$ posūkio matricomis.

Skaičiavimams paskiriami tokie kampai:

- Keičiami kampai (kintamieji uždavinys) sprendžiant uždavinį daugiau negu vieną kartą, pirmo segmento apie y ašį kampas γ_{1i} ir kampas apie z ašį δ_{1i} , trečio segmento kampas apie y ašį γ_{3i} ir kampas apie x ašį σ_{3i} , (čia indeksas i žymi išsilenkimo varianto numerį).
- Fiksuoto dydžio kampai, kurie nurodomi arba gaunami uždavinio pradžioje pirmojo skaičiavimo metu ir daugiau nekinta: ϕ_1 – užduoties taško kampas apie z ašį, α – krypties vektoriaus pasukimo kampas apie y ašį, β – krypties vektoriaus pasukimo kampas apie z ašį, ρ_{1i} – pirmo segmento i -jo išsilenkimo varianto kampas pasukantis spindulį į x ašį 90° (priklausomai nuo varianto šis kampas teigiamas arba ne), ρ_{3i} – trečio segmento i -jo išsilenkimo varianto kampas pasukantis spindulį 90° apie y ašį (priklausomai nuo varianto šis kampas – teigiamas arba ne).
- Gaunami kiekvieno skaičiavimo metu kampai: γ_{2e} – antro segmento e -ojo išsilenkimo varianto segmento sulenkimo kampas tarp dviejų spindulių apie y ašį, γ_{2ei} – antro segmento e -ojo išsilenkimo varianto segmento i -ojo spindulio pasukimo kampas apie y ašį, γ_{2esi} – antro segmento e -ojo išsilenkimo varianto segmento i -ojo galo kryptį nurodančio vienetinio vektoriaus pasukimo kampas apie y ašį, χ_{2i} – antrojo segmento i -ojo vektoriaus esančio tarp pirmo ir trečio segmentų, kurio ilgis l_i , kampas apie y ašį, ρ_{2ei} – antro segmento e -jo varianto i -tojo spindulio pasukimo kampas apie x ašį, ω_{jei} – j -ojo segmento e -ojo varianto i -ojo vienetinio vektoriaus rodančio segmento galo kryptį pasukimo kampas apie y ašį. Toliau remiantis 2.4 poskyryje išspėstu uždaviniu atliekami skaičiavimai tokia tvarka.

Gaunami pirmo ir trečio segmentų išsilenkimo variantų R_{11} , R_{12} , R_{31} ir R_{32} spinduliai su formule (20). Tada gaunamos pirmo segmento 2-jų išsilenkimo variantų galo taško koordinatės per formules (48)(49)(50). Paskaičiuojamos trečio segmento dviejų išsilenkimo variantų galo taško koordinatės taikant formules (51)(52)(53).

$$W_{1i} = (R_{1i} \cdot (0, 0, 1)) \cdot R_{y, \rho_{1i}} \cdot R_{z, \delta_{1i}} \quad (48)$$

$$U_{1i} = (R_{1i} \cdot (-1, 0, 0)) \cdot R_{y, \gamma_{1i}} \cdot R_{z, \delta_{1i}} \quad (49)$$

$$P_{1i} = W_{1i} + U_{1i} \quad (50)$$

čia W_{1i} – pirmo segmento i -ojo išsilenkimo varianto spindulys pasuktas į x ašį kampu ρ_{1i} , o po to apie z ašį kampu, δ_{1i} , U_{1i} – pirmo segmento i -ojo išsilenkimo varianto spindulys pasuktas apie y ašį kampu γ_{1i} ir apie z ašį kampu, δ_{1i} , P_{1i} – pirmo segmento i -ojo išsilenkimo varianto segmento galo taškas.

$$W_{3i} = (R_{3i} \cdot (-1, 0, 0)) \cdot R_{y, \rho_{3i}} \cdot R_{x, \sigma_{3i}} \cdot R_{y, \alpha} \cdot R_{z, \beta} \quad (51)$$

$$U_{3i} = (R_{3i} \cdot (0, 0, 1)) \cdot R_{y, \gamma_{3i}} \cdot R_{x, \sigma_{3i}} \cdot R_{y, \alpha} \cdot R_{z, \beta} \quad (52)$$

$$P_{3i} = W_{3i} + U_{3i} \quad (53)$$

čia W_{3i} – pirmo segmento i -ojo išsilenkimo varianto spindulys pasuktas į z ašį kampu ρ_{3i} , po to apie x ašį kampu σ_{3i} , dar kartą apie y ašį, kampu α ir apie z ašį kampu β , U_{3i} – pirmo segmento i -ojo išsilenkimo varianto spindulys pirmiausiai pasuktas apie y ašį kampu γ_{3i} , o po to pakartojama W_{3i} taško pasukimų seka nuo antros posūkio matricos, P_{3i} – trečio segmento i -ojo išsilenkimo varianto segmento galo taškas.

Toliau randami 4-ių atkarpų ilgiai tarp 1 ir 3 segmentų išsilenkimo variantų pagal formulę (54).

$$I_i = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (54)$$

čia I_i – i-osios atkarpos ilgis, x_1, y_1, z_1 – pirmo taško koordinatės, x_2, y_2, z_2 – antro taško koordinatės.

Toliau tęsiami skaičiavimai tik su tais antro segmento išsilankstymo variantais, kurių atkarpų ilgiai yra trumpesni už iš anksto nurodytą antro segmento stuburo ilgį L_2 . Pirmiausia surandami vektorių, atitinkančių atrinktas tinkamas atkarpas, kampai χ_{2i} , kurie dar žymimi pagal antro segmento išsilankstymo varianto numerį e , kai e lyginis skaičius – $\chi_{\frac{e}{2}}$ ir kai e nelyginis skaičius – $\chi_{\frac{e+1}{2}}$. Šie kampai surandami iš stačiakampių trikampių skaičiuojant tangentą 2.4 poskyryje nurodytu būdu.

Toliau surandami antro segmento e -ojo išlenkimo varianto segmento sulenkimo kampai γ_{2e} apie ašį y pagal (33) formulę ir pritaikius kurią nors iš 2.4 poskyryje nurodytų metodų (Niutono metodas arba funkcijos skleidimas Teiloro eilute). Pagal (20) formulę suskaičiuojami antro segmento e -ųjų išsilankstymo variantų spindulių ilgiai R_e .

Toliau antro segmento e -ųjų išsilankstymo variantų i -ųjų galų (antro segmento pradžios arba pabaigos galas) kryptis rodantys vienetiniai vektoriai pasukami kampais γ_{2esi} (γ_{2esi} yra kampas, sudarytas iš atitinkamo $\chi_{\frac{e}{2}}$ arba $\chi_{\frac{e+1}{2}}$ kampo ir $\frac{\gamma_{2e}}{2}$ arba $90^\circ - \frac{\gamma_{2e}}{2}$ kampo sudėties arba atimties galimai papildomai pridėdant arba atimant 90° kampą, kuris kiekvienam antro segmento e -ojo išsilankstymo varianto i -ajam spinduliui užrašytas pagal jų grafinį vaizdą XY plokštumoje (žr. (41)-(44) formulių sudarymo aiškinimą)) apie y ašį ir kampais ρ_{2ei} apie x ašį pagal formulę (55). Kampas ρ_{2ei} gaunamas kiekvienam antro segmento e -jam išsilankstymo variantui atskirai, atėmus iš trečio segmento išsilankstymo varianto galo taško P_{3i} pirmo segmento galo tašką P_{1i} , kuriuos jungia e -asis antrojo segmento išsilankstymo variantas ir paėmus gautojo vektoriaus y ir z koordinates paskaičiuojant kampą ρ_{2ei} pagal stačiakampio trikampio tangento formulę. Kiekvienam antro segmento e -jam išsilankstymo varianto atvaizdavimui trimatėje erdvėje yra reikalinga sužinoti bent vieno spindulio taško koordinates, kurios yra gaunamos pagal formulę (56). Čia γ_{2ei} yra spindulio antro segmento e -ojo išsilankstymo varianto pasukimo kampas apie y ašį, kuris, priklausomai nuo spindulio varianto, gaunamas prie kampo γ_{2esi} pridėjus arba atėmus 90° .

$$K_{ei} = (1,0,0) \cdot \mathbf{R}_{y,\gamma_{2esi}} \cdot \mathbf{R}_{x,\rho_{2ei}} \quad (55)$$

$$W_{2ei} = \left((R_e \cdot (1,0,0)) \cdot \mathbf{R}_{y,\gamma_{2ei}} \cdot \mathbf{R}_{x,\rho_{2ei}} \right) + P_{ji} \quad (56)$$

čia K_{ei} – antro segmento e -ojo išsilankstymo varianto i -ojo galo vienetinio vektoriaus taško koordinatės, W_{2ei} – antro segmento e -ojo išsilankstymo varianto i -ojo spindulio taškas, P_{ji} – j segmento ((56) formulės atveju 1-o ar 3-io segmento) i -ojo varianto, segmento galo taško koordinatės.

Surandami likę nežinomi pirmojo ir trečiojo segmentų 1-ą ir 2-ą išsilankstymo variantų kryptis nurodantys vienetiniai vektoriai pagal formules (57)(58)(59)(60).

$$J_{11} = (1,0,0) \cdot \mathbf{R}_{y,-\gamma_{11}+90^\circ} \cdot \mathbf{R}_{z,\delta_{11}} \quad (57)$$

$$J_{12} = (-1,0,0) \cdot \mathbf{R}_{y,450^\circ-(180^\circ-\gamma_{12})} \cdot \mathbf{R}_{z,\delta_{12}} \quad (58)$$

$$J_{31} = (1,0,0) \cdot \mathbf{R}_{y,\alpha+180^\circ+\gamma_{31}} \cdot \mathbf{R}_{x,\sigma_{31}} \cdot \mathbf{R}_{y,\alpha} \cdot \mathbf{R}_{z,\beta} \quad (59)$$

$$J_{32} = (1,0,0) \cdot R_{y,\alpha+180^\circ-\gamma_{31}} \cdot R_{x,\sigma_{32}} \cdot R_{y,\alpha} \cdot R_{z,\beta} \quad (60)$$

čia J_{11} ir J_{12} – 1-o segmento 1-o ir 2-o variantų segmento išsilenkimo variantų vienetiniai vektoriai, parodantys segmentų galų kryptis, J_{31} ir J_{32} – 3-io segmento 1-o ir 2-o variantų segmento išsilenkimo variantų vienetiniai vektoriai, parodantys segmentų galų kryptis.

Toliau gauti antrų segmentų e-ųjų išsilenkimo variantų vienetiniai kryptį nurodantys vektoriai K_{ei} palyginami su atitinkamais pirmo ir trečio segmentų vienetiniais vektoriais gautais iš formulių (57)(58)(59)(60) taikant vektorinės sandaugos formulę (45) ir vektorių lygiagretumo nesutapimo paklaidą išreiškus per vektoriaus ilgio skaičiavimo formulę (46). Taip pat patikrinama, ar lyginamų vienetinių krypties vektorių kryptys yra priešingos su formule (47).

Iš reikalavimus atitinkančio roboto išsilankstymo varianto, kuris sudarytas iš i-ojo pirmo segmento, e-tojo antro segmento ir i-ojo trečio segmento, išsireiškiama ieškomi kiekvieno segmento sulenkimo kampai θ_{gj} apie y ašis bei pasukimo kampai ϕ_{gj} apie z ašis.

Antro segmento kampas ϕ_{g2} yra apskaičiuojamas suradus vektorių G_{2e} pagal formulę (61) ir iš gauto vektoriaus koordinatčių kampą paskaičiavus su tokia stačiakampio trikampio tangento formule (62).

$$G_{2e} = (P_{3i} - P_{1n}) \cdot R_{z,-\delta_{1n}} \cdot R_{y,\gamma_{1n}} \quad (61)$$

$$\phi_{g2e} = \tan^{-1} \frac{|y_{G_{2e}}|}{|x_{G_{2e}}|} \quad (62)$$

čia P_{3i} – trečiojo segmento i-ojo išsilenkimo varianto galo taško koordinatės, P_{1n} – pirmojo segmento i-ojo išsilenkimo varianto galo taško koordinatės, $y_{G_{2e}}$ – G_{2e} vektoriaus y koordinatė, $x_{G_{2e}}$ – G_{2e} vektoriaus x koordinatė.

Trečio segmento kampas ϕ_{g3} yra apskaičiuojamas suradus vektorių G_{3i} pagal formulę (63) ir iš gauto vektoriaus koordinatčių kampą paskaičiavus su tokia stačiakampio trikampio tangento formule (64).

$$G_{3i} = (A - P_{3i}) \cdot R_{z,-\delta_{1n}} \cdot R_{y,\gamma_{1n}} \cdot R_{z,-\phi_{g2}} \cdot R_{y,-\gamma_{2e}} \quad (63)$$

$$\phi_{g3i} = \tan^{-1} \frac{|y_{G_{3i}}|}{|x_{G_{3i}}|} \quad (64)$$

čia A – užduoties taško, kuris buvo pasuktas į XZ plokštumą, koordinatės, P_{3i} – trečiojo segmento i-ojo išsilenkimo varianto galo taško koordinatės, $y_{G_{3i}}$ – G_{3i} vektoriaus y koordinatė, $x_{G_{3i}}$ – G_{3i} vektoriaus x koordinatė.

Atrinkto i-ojo pirmo segmento kampų atsakymai:

$$\theta_{g1} = \gamma_{1i} \quad (65)$$

$$\phi_{g1} = |\phi_1 - \delta_{1i}| \quad (66)$$

Atrinkto e-ojo antro segmento kampų atsakymai:

$$\theta_{g2} = \gamma_{2e} \quad (67)$$

$$\phi_{g2} = \phi_{g2e} \quad (68)$$

čia nustatčius kuriame XY koordinačių plokštumos ketvirtyje yra G_{2e} vektorius pagal tai prie ϕ_{g2} pridedama 0° , 90° , 180° arba 270° .

Atrinkto i-ojo trečio segmento kampų atsakymai:

$$\theta_{g3} = \gamma_{3i} \quad (69)$$

$$\phi_{g3} = \phi_{g3i} \quad (70)$$

čia nustatčius kuriame XY koordinačių plokštumos ketvirtyje yra G_{3i} vektorius pagal tai prie ϕ_{g3} pridedama 0° , 90° , 180° arba 270° .

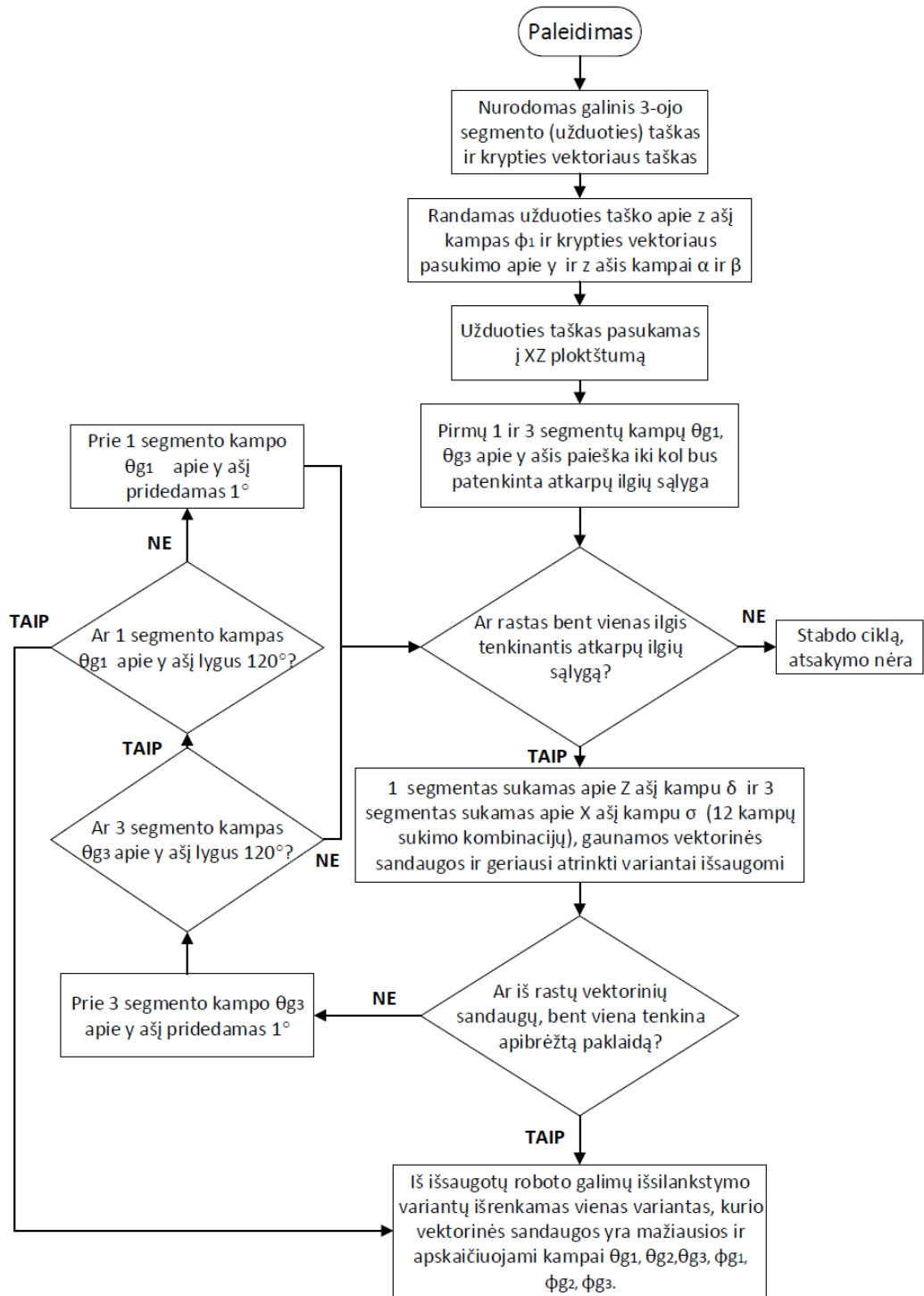
Pagal šiame skyriuje išvardintas skaičiavimo formules bei sudarytą skaičiavimo eigą, buvo sukurtas straublio tipo roboto 3-jų segmentų valdymo algoritmas, kurio veikimo schema yra pavaizduota 32 pav. Pagal algoritmą parašyta straublio tipo roboto 3-jų segmentų valdymo programa, naudojant programinę įrangą „Matlab“.

Pirmiausia, roboto valdymo programoje nurodomi straublio tipo roboto visų 3-jų segmentų ilgiai. Toliau, nurodžius užduoties taško, kurį reikia pasiekti robotui, koordinates ir vienetinio vektoriaus, rodančio trečio segmento galo kryptį, koordinates yra paleidžiama programa. Randamas užduoties taško pasisukimo kampas apie z ašį, kuris naudojamas pasukti užduoties tašką į XZ plokštumą ir vienetinio krypties vektoriaus kampai apie y ir z ašis. Šie kampai vėliau naudojami skaičiavimuose, kai trečią segmentą sulenkus apie y ašį ir pasukus apie z ašį, reikia orientuoti trečią segmentą pagal iš anksto nurodyto vienetinio krypties vektoriaus kryptį.

Pasukus užduoties tašką į XZ plokštumą, toliau yra ieškomi pirmo ir trečio segmento išsilenkimo variantus jungiančių atkarpų ilgiai (žr. 28 pav.). Atkarpų ilgiai ieškomi tol, kol visos atkarpos yra ilgesnės už iš anksto apibrėžtą sąlygoje antro segmento stuburo ilgį arba tol, kol pirmo ir trečio segmentų sulenkimo kampai abu nepasiekė maksimalaus leistino segmento sulenkimo kampo, tai yra, 120° . Ieškant atkarpų ilgių yra keičiami tiek pirmo, tiek trečio segmentų sulenkimo kampai apie y ašis, tiek segmentų pasukimo kampai apie z ašis. Atkarpų ilgių sąlyga yra patenkinama tik tada, kai bent vienos atkarpo ilgis yra randamas trumpesnis už iš anksto apibrėžtą antro segmento stuburo ilgį. Tokio ilgio neradus, programa yra stabdoma, nes nėra galimo roboto išsilenkimo varianto.

Kai yra patenkinama atkarpų ilgių sąlyga, toliau pirmą segmentą sukant apie z ašį, o trečią segmentą apie x ašį (iš trečio segmento pabaigos galo žiūrint gaunasi, jog suka apie trečio segmento z ašį) yra ieškomi antro segmento išsilenkimo variantai ir randami kiekvieno rasto antro segmento išsilenkimo varianto segmento galus išreiškiantys vienetiniai vektoriai. Šie rasti antro segmento vienetiniai vektoriai yra sulyginami su atitinkamais pirmo arba trečio segmento išsilenkimo varianto galų nurodančiais vienetiniais vektoriais, pritaikant vektorinių ir skaliarinių vektorių sandaugų savybes. Geriausių gautų variantų roboto išsilankstymo variantų segmentų kampai apie y ir z ašis bei roboto išsilankstusių roboto segmentų priekio ir galo taško koordinatės yra išsaugomos. Tada patikrinama, ar tarp išsaugotų roboto išsilankstymo variantų yra toks variantas, kurio pirmo, antro segmentų ir antro, trečio segmentų galų kryptis nurodančių vienetinių vektorių vektorinių sandaugų suma neviršija norimos ribos (šiuo atveju turi neviršyti 0,08). Jeigu taip, tai uždavinys yra baigiamas suskaičiuojant geriausio atrinkto varianto segmentų kampus apie y ir z ašis pagal

(65)(66)(67)(68)(69)(70) formules. Jeigu ne, tada keičiami 3-o segmento, o vėliau ir 1-o segmento y ašies kampai tol, kol randamas toks roboto išsilankstymas, kur vektorinių sandaugų sumų paklaidos neviršija nustatytos 0,08 ribos arba kol pirmo ir trečio segmento kampai apie y ašį abu netampa lygūs 120° . Po to algoritmas baigiamas suskaičiuojant geriausio atrinkto varianto segmentų kampus θ_{g1} , θ_{g2} , θ_{g3} apie y ašį ir kampus ϕ_{g1} , ϕ_{g2} , ϕ_{g3} apie z ašį.



32 pav. Straublio tipo roboto 3-jų segmentų valdymo algoritmas pritaikius vektorius, ieškantis kampų nuosekliu paieškos metodu

3. Straublio tipo roboto valdymo algoritmo tyrimo rezultatai

3.1. Valdymo algoritmo greitimeikos tyrimo rezultatai

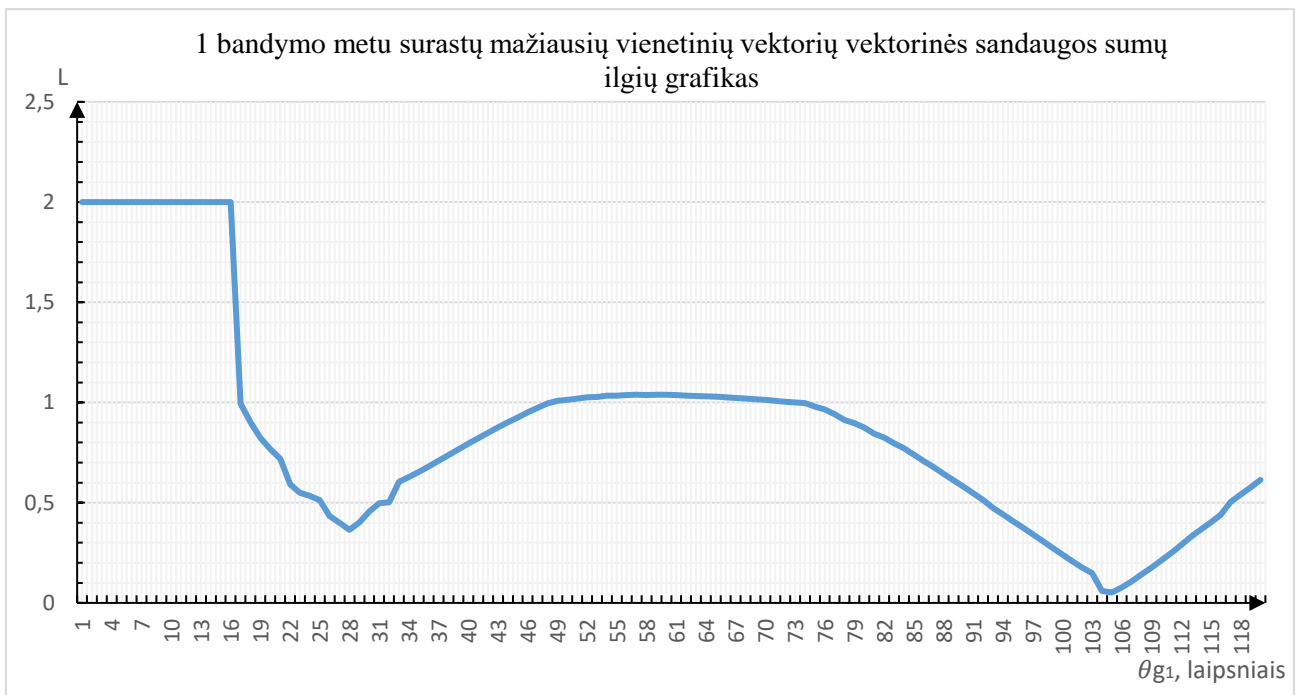
Paskutiniame šio magistro darbo uždavinyje yra nurodyta, jog reikia įvertinti sukurto straublio tipo roboto valdymo algoritmo (žr. 32 pav.) greitimeiką ir tikslumą. Pirmiausia buvo vertinama greitimeika. Iš anksto nurodžius skirtingus užduoties taškus, kuriuos reikia pasiekti robotui ir iš jų einančių vienetinių vektorių taškus, rodančius straublio tipo roboto 3-io segmento galo kryptį, buvo paleidžiama sukurta roboto valdymo programa ir matuojamas laikas bei skaičiuojamos vektorių vektorinės ir skaliarinės sandaugų iteracijos (1 iteracija yra 2 vektorinės sandaugos ir 2 skaliarinės sandaugos – vienas galimas pilnas roboto išsilankstymas). Sudarytas straublio tipo roboto valdymo programos kodas ieško geriausio roboto išsilankstymo nuosekliu paieškos metodu, todėl programos kode yra įvesta sąlyga, jog kodas yra stabdomas, jeigu yra surandamas straublio tipo roboto išsilankstymo variantas, kurio abiejų vektorių sandaugų ((45) formulė) vektorių ilgių ((46) formulė) suma yra mažesnė negu 0,08 (vienetinio vektoriaus ilgio matavimo vienetai). Kuo rasta vektorių sandaugų vektorių ilgių suma artimesnė 0, tuo tikslesnis rastas roboto išsilankstymas. Priešingu atveju, nerandant tokio varianto programos kodas veikia tol, kol pirmo ir trečio segmento sulenkimo kampai apie y ašis nepasiekia 120° . Greitimeika vertinta atlikus 10 bandymų su skirtingomis užduoties taško koordinatėmis ir kryptį nurodančių vienetinių vektorių koordinatėmis (žr. 1 lentelę). Atliekant 1 lentelėje esančius bandymus, roboto segmentų visi trys ilgiai yra vienodi ir lygūs 40 cm.

1 lentelė. Straublio tipo roboto 3-jų segmentų valdymo algoritmo skaičiavimo laikai ir iteracijos, valdymo algoritmui ieškant atsakymų nuosekliu paieškos metodu

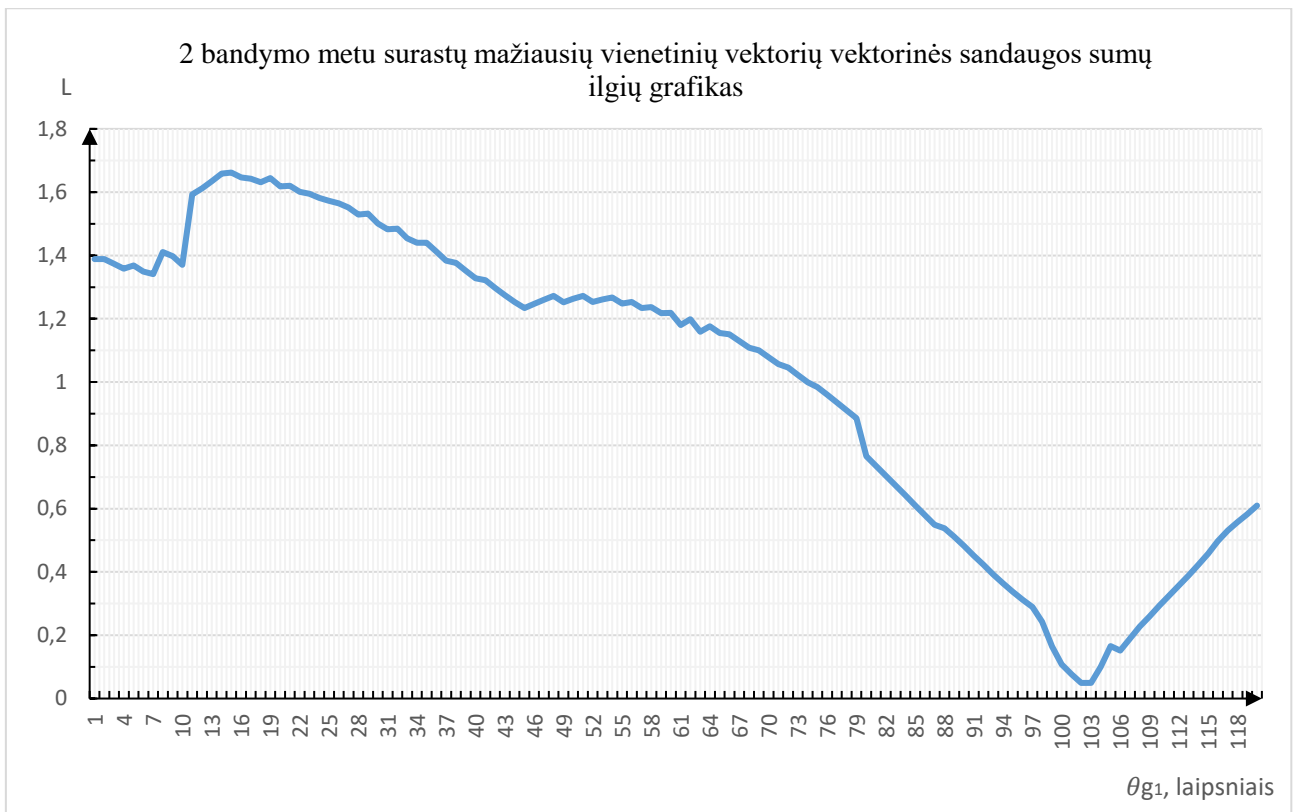
| Bandymo nr. | Užduoties taško koordinatės (x, y, z), cm | Vienetinio krypties vekt. koordinatės (x, y, z), cm | Algoritmo laikas, s | Iteracijų skaičius | Surasta mažiausia vektorinės sandaugos ilgių suma | Pirmo segmento kampas θ_{g1} apie y ašį, laipsniais |
|-------------|---|---|---------------------|--------------------|---|--|
| 1 | 56,8154; 51,8693; 67,7102 | 57,6683; 52,3617; 67,5366 | 616,2479 | 249884 | 0,0773 | 104 |
| 2 | -34,8668; -3,0505; 78,0000 | -35,6681; -3,2208; 78,5736 | 293,8640 | 105078 | 0,0606 | 102 |
| 3 | -10,7510; 27,0074; 68,0000 | 9,9103; 28,7663; 67,5152 | 512,3145 | 215654 | 0,1148 | 102 |
| 4 | 27,9596; 41,4519; 88,0000 | 28,6996; 41,2536; 88,6428 | 498,2314 | 164112 | 0,1257 | 94 |
| 5 | 55,7531; -15,9870; 77,0000 | 56,6608; -16,2302; 76,6580 | 233,6744 | 73262 | 0,0398 | 6 |
| 6 | 66,7450; 5,8394; 90,0000 | 67,4517; 5,8148; 90,7071 | 115,1618 | 37457 | 0,0760 | 21 |

| | | | | | | |
|----|----------------------------------|----------------------------------|----------|-------|--------|----|
| 7 | -63,8908; 48,1452; 40,0000 | -63,5501; 48,1154; 39,0603 | 67,3684 | 17119 | 0,0762 | 41 |
| 8 | 10,8800; 8,8105; 104,0000 | 10,9149; 8,8105; 104,9994 | 181,8145 | 47850 | 0,0772 | 37 |
| 9 | 47,5380; -44,3299; 40,0000 | 48,1808; -44,3299; 39,2340 | 73,9126 | 17638 | 0,0432 | 6 |
| 10 | 55,1774; 44,6817; 78,0000 | 55,0382; 44,6817; 78,9903 | 49,1726 | 15616 | 0,0581 | 35 |

32 paveikslėlyje pavaizduoto valdymo algoritmo pirmo segmento kampas θ_{g1} apie y ašį sukurtoje valdymo programoje yra išreikštas kaip pats pagrindinis „while“ ciklas, kuriame yra visi kiti ciklai ir programa baigiama tik radus vienetinių vektorių vektorinės sandaugos ilgių sumą mažesnę už 0,08 arba programa stabdoma, kai pirmo segmento sulenkimo kampas θ_{g1} apie y ašį pasiekia 120° . Todėl prie kiekvieno pirmo segmento sulenkimo kampo θ_{g1} laipsnio apie y ašį yra gaunama išrinkta mažiausia vienetinių vektorių vektorinės sandaugos ilgių suma L . Atsižvelgiant į tai, pirmo segmento sulenkimo kampas θ_{g1} išreiškiamas intervalu nuo 0° iki 120° (pagal pirmo roboto segmento sulenkimo ribas) ir šiame intervale išreiškiami surasti mažiausi vienetinių vektorių vektorinės sandaugos ilgiai. 33 ir 34 paveikslėliuose yra pavaizduoti iš 1-os lentelės 1-o ir 2-o bandymų surasti mažiausi vienetinių vektorių vektorinės sandaugos ilgiai kampui θ_{g1} kintant nuo 0° iki 120° .



33 pav. 1 bandymo surasti mažiausi vienetinių vektorių vektorinės sandaugos ilgiai kampui θ_{g1} kintant nuo 0° iki 120°



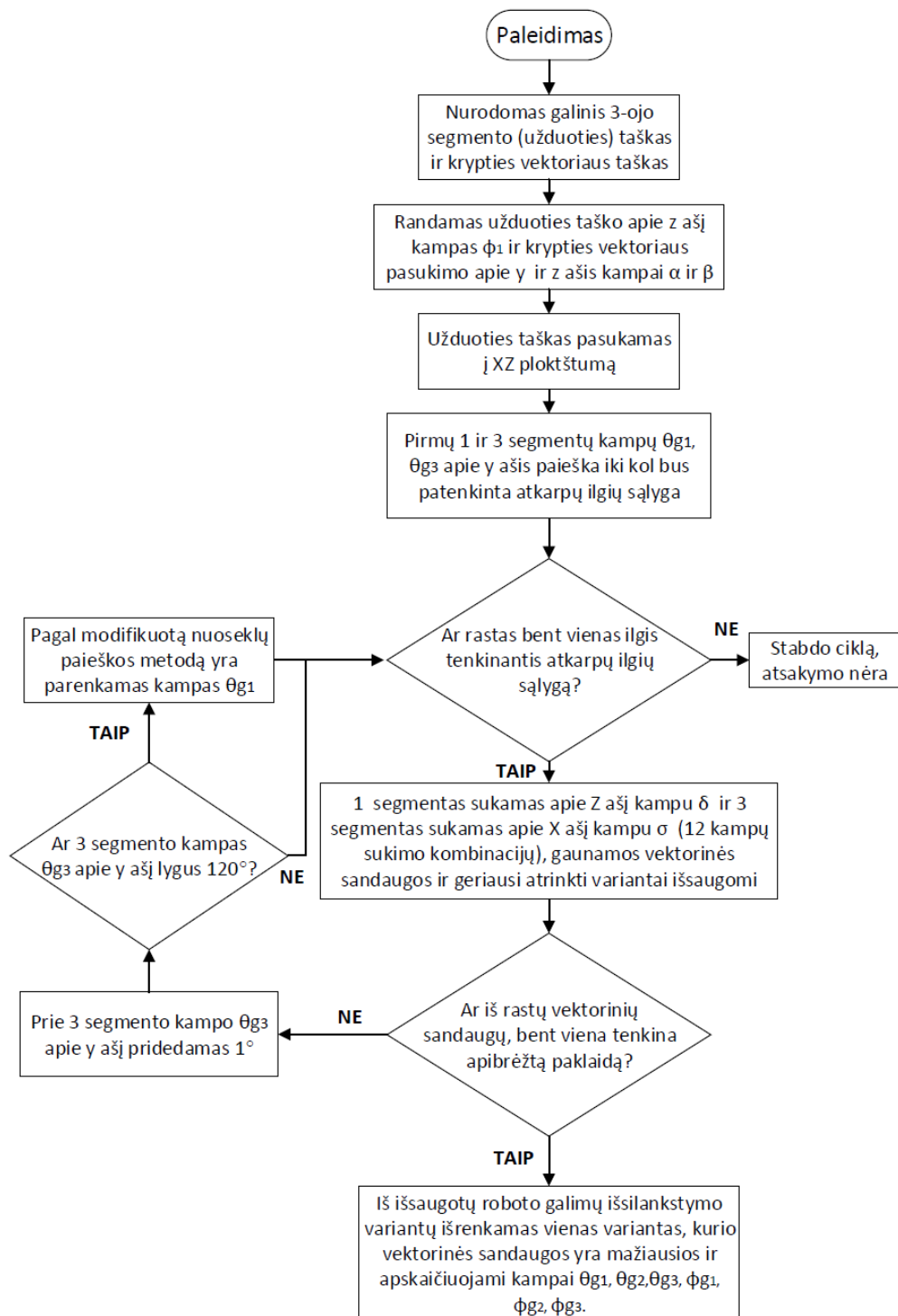
34 pav. 2 bandymo surasti mažiausi vienetinių vektorių vektorinės sandaugos ilgiai kampui θ_{g1} kintant nuo 0° iki 120°

Iš 32 ir 33 paveikslėlių matyti, jog surasti mažiausi vienetinių vektorių vektorinės sandaugos ilgiai keičiantis θ_{g1} kinta netiesiškai ir kiekviename iš pavaizduotų grafikų yra po du ekstremumus (vienas lokalus minimumas, o kitas globalus minimumas), kurių kitais bandymų atvejais galimai gali būti ir daugiau. Jeigu L išreikštume kaip funkciją $f(x)$, o θ_{g1} išreikštume kaip x , tai $f(x)$ būtų multimodinė funkcija. Kadangi roboto valdymo programos tikslas rasti tokius roboto segmentų išsilankstymo ir pasukimų kampus, kad rasta vektorių sandaugų vektorių ilgių suma artimesnė 0, tai galima pritaikyti paieškos metodą, kuris parinkintų kampą θ_{g1} ir jo nereiktų keisti nuosekliai pridėdant po 1° kaip pavaizduota 32 paveikslo sudarytam roboto valdymo programos algoritme. Taip yra siekiama optimizuoti straublio tipo roboto valdymo algoritmo skaičiavimo greitį.

Valdymo algoritmo greičiui optimizuoti, buvo modifikuotas nuoseklios paieškos metodas, papildomai pritaikant Dichotomijos paieškos metodą. Pagal šį modifikuotą nuoseklios paieškos metodą roboto valdymo programoje pirmiausia yra nustatomos intervalo ribos. Šios ribos nustatomos surandant pirmą ir paskutinį pirmo segmento sulenkimo kampą θ_{g1} , kai atstumas tarp pirmo ir trečio segmentų išsilankstymo yra mažesnis negu iš anksto nurodyto antro segmento stuburo ilgis, kuris šiuo atveju lygus 40 cm. Toliau modifikuojamas nuoseklios paieškos metodas, kampo θ_{g1} intervalą dalijant į n dalių nurodant kampo θ_{g1} žingsnio dydį laipsniais. Ieškomos mažiausios vektorinės sandaugos ilgių sumos nurodytuose taškuose žingsniu atgal nuo intervalo galo. Jeigu kažkuriame iš nurodytų taškų randama L vertė mažesnė negu 0,08, tada paieškos algoritmas stabdomas ir atliekamas paskutinis 32 paveiksle pavaizduoto algoritmo punktas.

Jeigu nurodytuose taškuose nerandama L reikšmė mažesnė už 0,08, tada imamas taškas su mažiausia rasta L reikšme intervale. Čia yra pritaikoma Dichotomijos paieškos metodo idėja. Surastas θ_{g1} su

mažiausia L reikšmė intervale padaromas naujo intervalo, prasidedančio ir pasibaigiančio tarp dviejų artimiausių prieš tai praėtų intervalo taškų, centru. Ši nauja sritis sudalinama jau į 4 lygias dalis, su naujais dviem θ_{g1} taškais ir apskaičiuojamos vektorinių sandaugų ilgių sumos juose. Jeigu vėl nerandamas L mažesnis už 0,08, naujo intervalo dvi θ_{g1} taškų reikšmės lyginamos tarpusavyje ir parenkama iš jų mažesnė reikšmė, kuri po to padaroma sekančio naujo intervalo centru, kuris vėl dalijamas į 4 dalis. Taip daroma tol, kol randama mažiausia L reikšmė apribotoje θ_{g1} intervalo srityje. Šis modifikuotas nuoseklios paieškos metodas buvo pritaikytas straublio tipo roboto valdymo algoritmui (žr. 32 pav.) ir buvo gautas toks roboto valdymo algoritmas (žr. 35 pav.).[31]



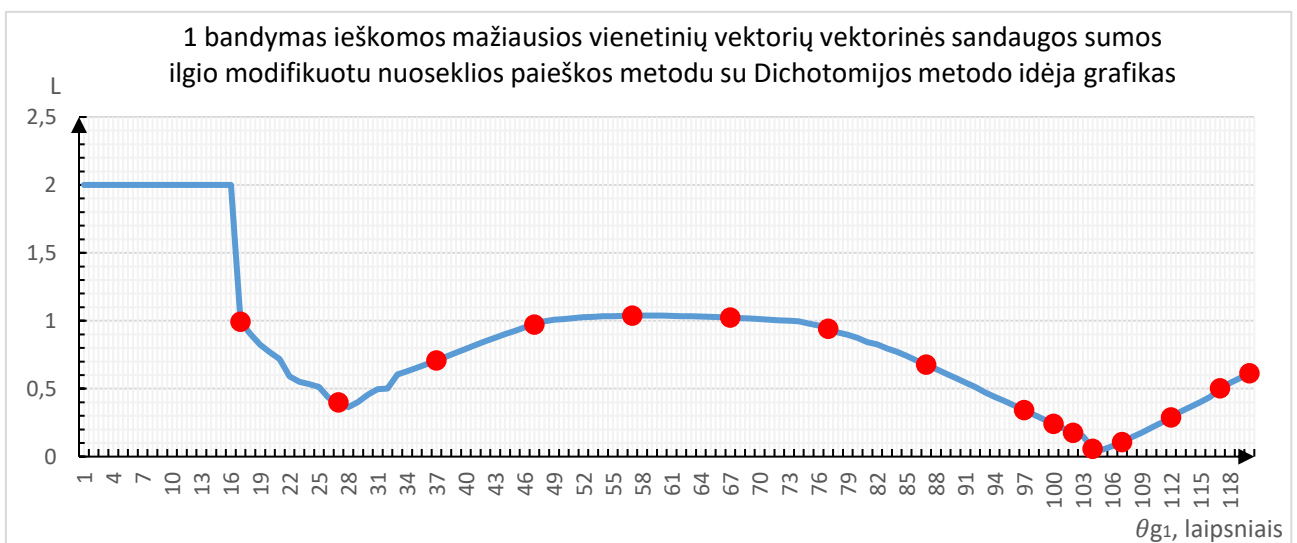
35 pav. Straublio tipo roboto 3-jų segmentų valdymo algoritmas pritaikius vektorius, ieškantis kampų modifikuotu nuosekliu paieškos metodu su Dichotomijos paieškos metodo idėja

Pritaikius 35 paveikslėlyje pavaizduotą roboto valdymo algoritmą, buvo pakartoti 10 bandymų su tais pačiais užduoties taškais ir vienetiniams krypties vektoriams. Gauti rezultatai atvaizduoti 2 lentelėje, kiekvienas bandymas pakartotas su skirtingais kampo θ_{g1} žingsniais. 1 ir 2 bandymų rezultatai, kai kampo žingsnis yra 10° , atvaizduoti 36 ir 37 paveiksluose.

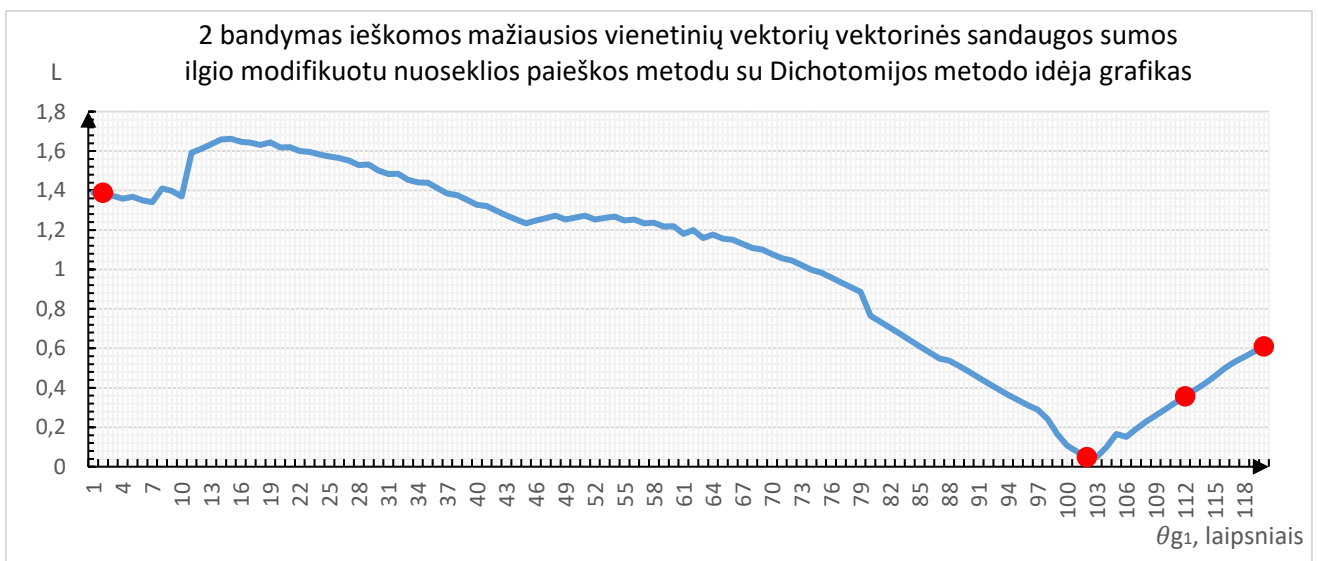
2 lentelė. Straublio tipo roboto 3-jų segmentų valdymo algoritmo skaičiavimo laikai ir iteracijos, valdymo algoritmui ieškant atsakymų modifikuotu nuoseklios paieškos metodu su Dichotomijos metodo idėja

| Bandymo nr. | Užduoties taško koordinatės (x, y, z), cm | Vienetinio krypties vekt. koordinatės (x, y, z), cm | Kampo θ_{g1} pradinis paieškos žingsnis, laipsniais | Algoritmo laikas, s | Iteracijų skaičius | Surasta mažiausia vektorinės sandaugos ilgių suma L | Pirmo segmento kampas θ_{g1} apie y ašį, laipsniais |
|-------------|---|---|--|---------------------|--------------------|---|--|
| 1 | 56,8154; 51,8693; 67,7102 | 57,6683; 52,3617; 67,5366 | 5 | 227,1136 | 83878 | 0,0773 | 104 |
| | | | 10 | 202,1399 | 76983 | 0,0773 | 104 |
| | | | 15 | 233,2183 | 87265 | 0,0773 | 104 |
| 2 | -34,8668; -3,0505; 78,0000 | -35,6681; -3,2208; 78,5736 | 5 | 41,0602 | 13031 | 0,0574 | 102 |
| | | | 10 | 26,2807 | 7681 | 0,0574 | 102 |
| | | | 15 | 139,2349 | 53265 | 0,0767 | 101 |
| 3 | -10,7510; 27,0074; 68,0000 | 9,9103; 28,7663; 67,5152 | 5 | 99,1244 | 26768 | 0,1154 | 101 |
| | | | 10 | 71,0450 | 19052 | 0,1154 | 101 |
| | | | 15 | 418,8102 | 139709 | 0,1220 | 54 |
| 4 | 27,9596; 41,4519; 88,0000 | 28,6996; 41,2536; 88,6428 | 5 | 174,2150 | 48763 | 0,1257 | 94 |
| | | | 10 | 120,1854 | 34733 | 0,1257 | 94 |
| | | | 15 | 124,1455 | 40999 | 0,1257 | 94 |
| 5 | 55,7531; -15,9870; 77,0000 | 56,6608; -16,2302; 76,6580 | 5 | 222,0468 | 69261 | 0,0398 | 6 |
| | | | 10 | 243,8049 | 73444 | 0,0398 | 6 |
| | | | 15 | 93,8736 | 26003 | 1,0226 | 68 |
| 6 | 66,7450; 5,8394; 90,0000 | 67,4517; 5,8148; 90,7071 | 5 | 75,0525 | 21820 | 0,0447 | 80 |
| | | | 10 | 261,3537 | 71209 | 0,0775 | 79 |
| | | | 15 | 102,6855 | 27512 | 0,0717 | 83 |
| 7 | -63,8908; 48,1452; 40,0000 | -63,5501; 48,1154; 39,0603 | 5 | 109,4419 | 27406 | 0,0757 | 41 |
| | | | 10 | 47,7543 | 14089 | 0,0479 | 42 |
| | | | 15 | 36,6759 | 11440 | 0,0479 | 42 |

| | | | | | | | |
|----|----------------------------------|----------------------------------|----|----------|-------|--------|----|
| 8 | 10,8800; 8,8105; 104,0000 | 10,9149; 8,8105; 104,9994 | 5 | 90,4467 | 24814 | 0,0694 | 61 |
| | | | 10 | 51,6017 | 11854 | 0,0694 | 61 |
| | | | 15 | 43,2637 | 7050 | 0,0694 | 61 |
| 9 | 47,5380; -44,3299; 40,0000 | 48,1808; -44,3299; 39,2340 | 5 | 85,3961 | 26573 | 0,0278 | 7 |
| | | | 10 | 52,4539 | 15637 | 0,0278 | 7 |
| | | | 15 | 55,6539 | 15991 | 0,0432 | 6 |
| 10 | 55,1774; 44,6817; 78,0000 | 55,0382; 44,6817; 78,9903 | 5 | 54,7928 | 17700 | 0,0632 | 93 |
| | | | 10 | 141,1295 | 55496 | 0,0764 | 91 |
| | | | 15 | 78,5340 | 29402 | 0,0723 | 92 |



36 pav. 1 bandymas ieškamos mažiausios vienetinių vektorių vektorinės sandaugos sumos ilgio modifikuotu nuoseklios paieškos metodu grafikas, pradinis paieškos žingsnis – 10°



37 pav. 2 bandymas ieškamos mažiausios vienetinių vektorių vektorinės sandaugos sumos ilgio modifikuotu nuoseklios paieškos metodu grafikas, pradinis paieškos žingsnis – 10°

Poskyrio išvados ir rezultatai:

- Ištirta straublio tipo roboto valdymo programos greitimeika atliekant 10 bandymų su skirtingomis užduoties koordinatėmis ir vienetinių krypčių vektoriais. Vidutinis programos skaičiavimo laikas šių bandymų metu yra 264,1762 sekundės (apie 4 minutes 24 sekundes).
- Siekiant optimizuoti programos skaičiavimo greitį, skaičiavimuose pirmo segmento sulenkimo kampui θ_{g1} parinkti buvo pritaikytas modifikuotas nuoseklios paieškos metodas su Dichotomijos paieškos metodo idėja. Buvo pakartota 10 bandymų su tomis pačiomis užduoties taškų koordinatėmis ir vienetiniais krypčių vektoriais su pradiniais kampo θ_{g1} žingsniais, kurie lygūs 5° , 10° ir 15° .
- Gauti tokie vidutiniai straublio tipo roboto valdymo programos skaičiavimo laikai, kai pradinis žingsnis 5° , skaičiavimo laikas – 117,869 sekundės (apie 1 minutę 58 sekundes), kai pradinis žingsnis 10° , skaičiavimo laikas – 121,775 sekundės (apie 2 minutes 2 sekundes) ir kai pradinis žingsnis 15° , skaičiavimo laikas – 132,610 sekundės (apie 2 minutes 13 sekundžių). Nors vidutinis skaičiavimo laikas yra mažiausias prie 5° pradinio žingsnio kampo, bet pastebėta, jog iš 10 bandymų, 8 greičiausiai buvo skaičiuojami prie 10° pradinio žingsnio kampo.
- Pastebėta, jog taikant modifikuotą nuoseklios paieškos metodą kampui θ_{g1} parinkti, buvo surasta artimesnių 0 vektorinių sandaugų ilgių sumų L . Taip yra dėl to, jog leidžiant valdymo programą ir ieškant roboto išsilankstymo nuosekliu paieškos metodu, kampas θ_{g1} yra pradedant nuo pirmo turimo θ_{g1} kampo keičiamas pridodant po 1° nuosekliai ir algoritmas yra stabdomas vos yra randama L reikšmė mažesnė už 0,08. Todėl esant keliems minimumams gali būti atveju, kai yra artimesnių 0 L reikšmių kampo θ_{g1} intervale, kuriuos randa programa su modifikuotu nuoseklios paieškos metodu arba pastarasis metodas priartėję prie 0 artimesnio rezultato iš kitos pusės, negu standartinis nuoseklios paieškos metodas.

3.2. Roboto išsilankstymo trimatėje erdvėje, pagal valdymo algoritmo surastų roboto segmentų pasukimo ir sulenkimo kampų dydžius, pozicionavimo tikslumo tyrimo rezultatai

Siekiant įvertinti straublio tipo roboto 3-jų segmentų, sukurtos valdymo programos, surasto roboto išsilankstymo varianto pozicijos ir orientacijos tikslumą, naudojant „Matlab“ programą, buvo pasirinkta susikurti 49 užduoties taškus, kuriuos turi pasiekti robotas, esančius ant 50 cm ilgio ir 50 cm pločio plokštumos (7x7 taškai dengia visą plokštumą).

Pirmiausia, kuriant užduoties taškus ant plokštumos, jie yra išdėliojami vienas nuo kito vienodais atstumais, XZ plokštumoje nuo taško (-25, 0, 25) iki taško (25, 0, -25). Pagal šiuos taškus sukuriama vienetinių vektorių taškai, kurie rodytų roboto segmento galo kryptį, sukūrus, tarpusavyje vienodais nutolusius, antrus 49 taškus XZ plokštumoje nuo taško (-25, 0, 25) iki taško (25, 0, -25), kurie visi pastumiami statmenai XZ plokštumai per 1 cm +y ašies kryptimi (nes vienetinių vektorių ilgis privalo būti lygus 1). Tada prie abiejų plokštumų visų taškų pridėjus 65 cm prie y koordinatę ir 60 cm prie z koordinatę abiejų plokštumų taškai perstumiami per 65cm nuo z ašies ir 60 cm nuo x ašies. Toliau visi abiejų plokštumų taškai padauginami iš posūkio matricių $R_{x,20^\circ}$ ir $R_{z,90^\circ}$, taip visus taškus pasukant 20° apie x ašį, o po to 90° apie z ašį. Gautos užduoties taškų plokštumos pasvirimo kampas tarp XY koordinatę plokštumos ir plokštumų yra apie $74,5^\circ$.

Turint 49-ių užduoties taškų koordinates ir 49-ių kryptį rodančių vienetinių vektorių taškų koordinates, šie taškai yra nurodomi sukurtai straublio tipo roboto valdymo programai ir yra

surandami straublio tipo roboto visų trijų segmentų sulenkimo kampai apie y ašis ir pasukimo kampai apie z ašis. Surasti, roboto geriausių išsilankstymų, 3-jų segmentų sulenkimo ir pasukimo kampai, prie pirmųjų 14 užduotų taškų (2 viršutinės taškų eilės iš 49 užduoties taškus turinčios plokštumos 7 eilių) koordinatinių atvaizduoti 3 lentelėje.

3 lentelė. Straublio tipo roboto 3-jų segmentų valdymo algoritmo surasti segmentų sulenkimo apie y ašis ir pasukimo apie z ašis kampai

| Užd. taško nr. | Nurodyto užduoties taško koord. (x, y, z), cm | Nurodyto vienetinio krypties vekt. koord. (x, y, z), cm | Pirmo segmento kampas θ_{g1} apie y ašį, laipsniais | Pirmo segmento kampas ϕ_{g1} apie z ašį, laipsniais | Antro segmento kampas θ_{g2} apie y ašį, laipsniais | Antro segmento kampas ϕ_{g2} apie z ašį, laipsniais | Trečio segmento kampas θ_{g3} apie y ašį, laipsniais | Trečio segmento kampas ϕ_{g3} apie z ašį, laipsniais |
|----------------|---|---|--|--|--|--|---|---|
| 1 | 90,1517; 25,0000; 57,6426 | 91,0914; 25,0000; 57,3005 | 65,0000 | 20,4992 | 9,5489 | 23,5096 | 60,0000 | 351,9338 |
| 2 | 90,1517; 16,6667; 57,6426 | 91,0914; 16,6667; 57,3005 | 94,0000 | 10,4742 | 70,666 | 4,9390 | 85,0000 | 347,7394 |
| 3 | 90,1517; 8,3334; 57,6426 | 91,0914; 8,3334; 57,3005 | 98,0000 | 5,2812 | 80,8408 | 3,0065 | 90,0000 | 353,1220 |
| 4 | 90,1517; 0,0000; 57,6426 | 91,0914; 0,0000; 57,3005 | 100,0000 | 0,0000 | 81,7971 | 0,0000 | 93,0000 | 0,0000 |
| 5 | 90,1517; -8,3334; 57,6426 | 91,0914; -8,3334; 57,3005 | 98,0000 | 353,7188 | 79,9896 | 358,4397 | 91,0000 | 8,2457 |
| 6 | 90,1517; -16,6667; 57,6426 | 91,0914; -16,6667; 57,3005 | 95,0000 | 348,5258 | 69,7449 | 356,8458 | 86,0000 | 13,2737 |
| 7 | 90,1517; -25,0000; 57,6426 | 91,0914; -25,0000; 57,3005 | 65,0000 | 339,5008 | 10,1249 | 336,1188 | 60,0000 | 7,1374 |
| 8 | 87,3016; 25,0000; 49,8118 | 88,2413; 25,0000; 49,4698 | 115,0000 | 15,9798 | 100,4776 | 5,4640 | 99,0000 | 349,0507 |
| 9 | 87,3016; 16,6667; 49,8118 | 88,2413; 16,6667; 49,4698 | 119,0000 | 10,8082 | 110,9918 | 3,7409 | 103,0000 | 352,6465 |
| 10 | 87,3016; 8,3334; 49,8118 | 88,2413; 8,3334; 49,4698 | 120,0000 | 5,4526 | 115,2634 | 2,9838 | 105,0000 | 356,1905 |

| | | | | | | | | |
|----|----------------------------------|----------------------------------|----------|----------|----------|----------|----------|---------|
| 11 | 87,3016; 0,0000; 49,8118 | 88,2413; 0,0000; 49,4698 | 120,0000 | 0,0000 | 115,7252 | 0,0000 | 105,0000 | 0,0000 |
| 12 | 87,3016; -8,3334; 49,8118 | 88,2413; -8,3334; 49,4698 | 120,0000 | 353,5474 | 114,2634 | 2,0161 | 104,0000 | 3,8095 |
| 13 | 87,3016; -16,6667; 49,8118 | 88,2413; -16,6667; 49,4698 | 118,0000 | 349,1918 | 112,3586 | 357,3240 | 103,0000 | 8,3073 |
| 14 | 87,3016; -25,0000; 49,8118 | 88,2413; -25,0000; 49,4698 | 113,0000 | 344,0202 | 100,4858 | 356,0092 | 101,0000 | 13,1817 |

Su 3 lentelėje surašytais kampais buvo sprendžiamas tiesioginis straublio tipo roboto kinematikos uždavinys, taip buvo ieškoma roboto galo taško koordinacių ir tuo pačiu roboto galo kryptį rodančio vienetinio krypties vektoriaus koordinacių. Tiesioginis kinematikos uždavinys buvo sprendžiamas su $R_{y,a}$ ir $R_{z,a}$ posūkio matricomis. Apskaičiuotos segmento galinio taško koordinatės nuo roboto bazės taško (0, 0, 0) ir segmento galo kryptį rodančio vienetinio vektoriaus koordinatės kiekvienam segmentui atskirai. Tada antro ir trečio segmentų galiniai taškai yra pasukami atitinkamai pagal pirmo segmento galo surasto vienetinio vektoriaus kryptį ir pagal pirmo, antro segmentų galų surastas vienetinių vektoriaus kryptis. Straublio tipo roboto galo taško koordinatės gaunamos sudėjus pirmo segmento galinio taško koordinatės su prieš tai surastomis antro ir trečio segmentų taškų, pasuktų pagal vienetinius krypties vektorius, koordinatėmis. Tokiu pačiu skaičiavimo principu gaunamos ir trečio segmento galo kryptį rodančio vienetinio vektoriaus koordinatės. 14-os straublio tipo roboto išsilankstymų, apskaičiuotų tiesioginiu kinematikos būdu roboto galinių taškų ir galo kryptį rodančių vienetinių vektorių, koordinatės pateiktos 4 lentelėje.

Siekiant įvertinti roboto galinio taško pozicionavimo paklaidas, buvo skaičiuojami euklidiniai atstumai $d(T_1, T_2)$ pagal (71) formulę. Pagal apskaičiuotus euklidinius atstumus tarp nurodytų užduoties taškų ir apskaičiuotų tiesioginiu kinematikos būdu roboto galinio taškų, gauname atkarpos ilgį, kuris parodo kiek robotas yra nutolęs nuo tikslo. Taip yra išreiškiama roboto pozicionavimo paklaida, kuo ji artimesnė 0, tuo robotas tiksliau pozicionuoja.

$$d(T_1, T_2) = \sqrt{(x_{T_2} - x_{T_1})^2 + (y_{T_2} - y_{T_1})^2 + (z_{T_2} - z_{T_1})^2} \quad (71)$$

čia $d(T_1, T_2)$ – euklidinis atstumas tarp 1-o ir 2-o taško, $x_{T_1}, y_{T_1}, z_{T_1}$ – pirmo taško koordinatės, $x_{T_2}, y_{T_2}, z_{T_2}$ – antro taško koordinatės.

Roboto trečio segmento orientavimo tikslumas įvertinamas apskaičiuavus nuokrypio kampą θ_N tarp nurodyto vienetinio krypties vektoriaus ir apskaičiuoto iš tiesioginės kinematikos uždavinio vienetinio krypties vektoriaus. Ieškant nuokrypio kampo, pirmiausiai gaunamos nurodyto ir apskaičiuoto vienetinių krypties vektorių, einančių iš taško (0, 0, 0), koordinatės, atėmus iš nurodyto vienetinio krypties vektoriaus koordinacių nurodytas užduoties taško koordinatės (4 lentelės 3 ir 2 stulpelių skirtumas) ir taip pat atėmus iš apskaičiuotų vienetinio krypties vektoriaus koordinacių apskaičiuotas roboto galo taško koordinatės (4 lentelės 5 ir 4 stulpelių skirtumas). Toliau gaunamos surastų vienetinių vektorių vektorinės sandaugos pagal (45) formulę ir apskaičiuojamos iš vektorių

sandaugų gautų vektorių ilgiai l pagal (46) formulę. Nuokrypio kampas θ_N apskaičiuojamas pagal (72) formulę. Šis kampas išreiškia surasto straublio tipo roboto išsilankstymo trečio segmento galo orientavimo paklaidą nuo 0° iki 180° . Kuo gautas θ_N kampas artimesnis 0° , tuo roboto galas trimatėje erdvėje yra orientuotas tiksliau. Apskaičiuotos išlankstyto, pagal 3 lentelėje pateiktus segmentų sulenkimo θ_{gj} ir pasukimų ϕ_{gj} kampus, straublio tipo roboto pozicionavimo ir orientavimo paklaidos pateiktos 4 lentelėje.

$$\theta_N = \sin^{-1} \left(\frac{l}{1} \right) \quad (72)$$

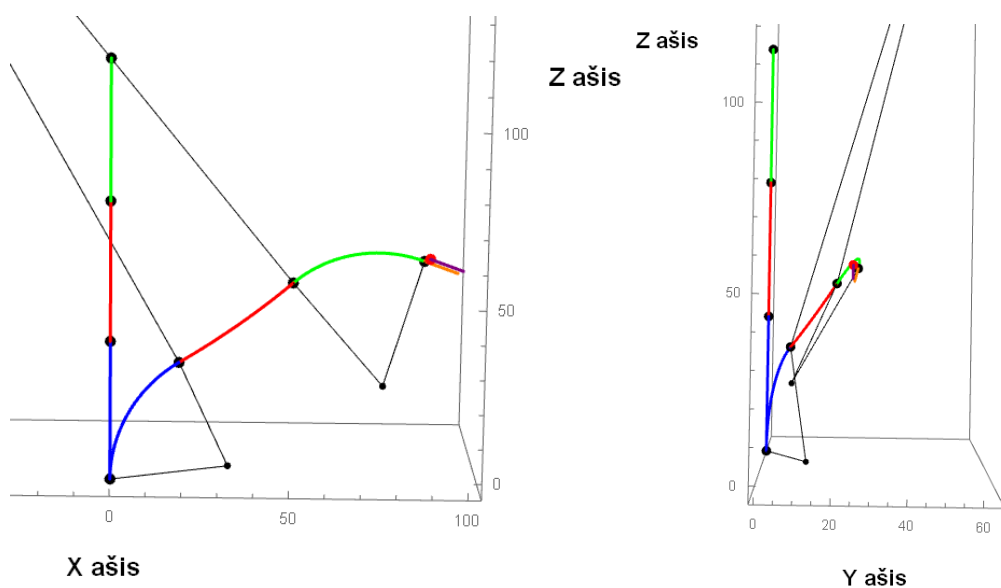
čia θ_N – nuokrypio kampas esantis tarp surasto roboto išsilankstymo varianto galo kryptį rodančio vienetinio vektoriaus ir iš anksto nurodyto vienetinio krypties vektoriaus, l – iš vektorinės sandaugos gauto vektoriaus ilgis.

4 lentelė. Straublio tipo roboto 3-jų segmentų valdymo algoritmo surasto geriausio išsilankstymo atvejo euklidinių atstumų ir vektoriaus nuokrypio kampų rezultatai

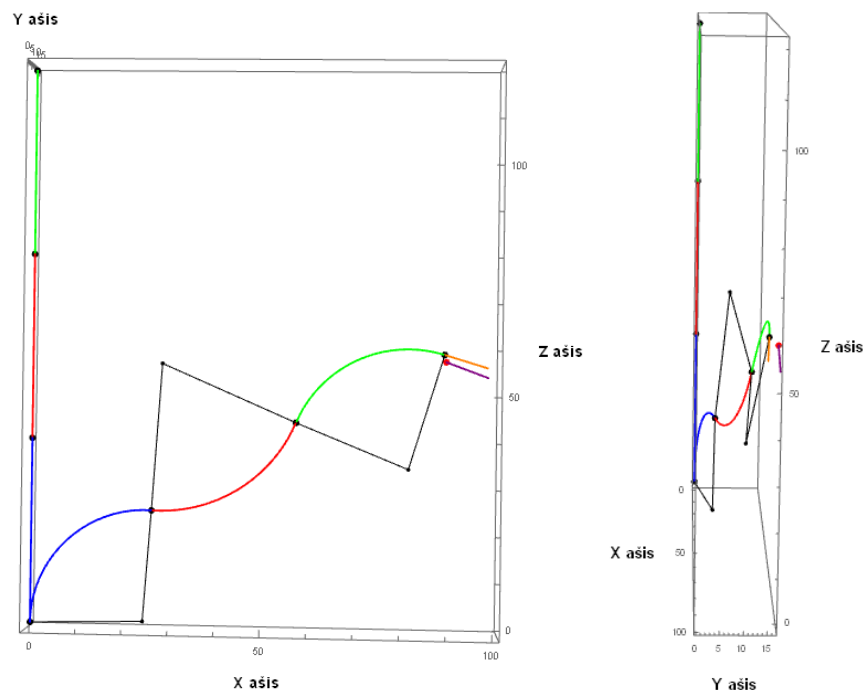
| Užd. taško nr. | Nurodyto užduoties taško koordinatės (x, y, z), cm | Nurodyto vienetinio krypties vekt. koordinatės (x, y, z), cm | Apskaičiuotos roboto galo taško koordinatės (x, y, z), cm | Apskaičiuoto vienetinio krypties vekt. koordinatės (x, y, z), cm | Surasta mažiausia vektorinės sandaugos ilgių suma | Euklidinis atstumas $d(T_1, T_2)$, cm | Vektoriaus nuokrypio kampas θ_N , laipsniais |
|----------------|--|--|---|--|---|--|---|
| 1 | 90,1517; 25,0000; 57,6426 | 91,0914; 25,0000; 57,3005 | 88,6055; 26,3151; 56,6897 | 89,5480; 26,2237; 56,3683 | 0,0784 | 2,2424 | 5,3707 |
| 2 | 90,1517; 16,6667; 57,6426 | 91,0914; 16,6667; 57,3005 | 89,6888; 14,7069; 59,2700 | 90,6451; 14,6483; 58,9835 | 0,0725 | 2,5891 | 4,7207 |
| 3 | 90,1517; 8,3334; 57,6426 | 91,0914; 8,3334; 57,3005 | 89,0410 7,5402 59,8382 | 89,9982 7,5002 59,5517 | 0,0626 | 2,5853 | 4,0484 |
| 4 | 90,1517; 0,0000; 57,6426 | 91,0914; 0,0000; 57,3005 | 90,7026; 0,0000; 56,6874 | 91,6349 0,0000 56,3258 | 0,0362 | 1,1027 | 1,1975 |
| 5 | 90,1517; -8,3334; 57,6426 | 91,0914; -8,3334; 57,3005 | 89,5220; -7,3648; 58,8617 | 90,4688; -7,3196; 58,5431 | 0,0669 | 1,6796 | 2,9463 |
| 6 | 90,1517; -16,6667; 57,6426 | 91,0914; -16,6667; 57,3005 | 90,8718; -14,7838; 56,7518 | 91,8111; -14,7259; 56,4136 | 0,0701 | 2,2039 | 3,3250 |
| 7 | 90,1517; -25,0000; 57,6426 | 91,0914; -25,0000; 57,3005 | 88,3937; -26,7050; 57,0937 | 89,3389; -26,6257; 56,7768 | 0,0735 | 2,5098 | 4,7783 |
| 8 | 87,3016; 25,0000; 49,8118 | 88,2413; 25,0000; 49,4698 | 87,7921; 25,2456; 48,7733 | 88,7148; 25,3178; 48,3948 | 0,0763 | 1,1744 | 4,7380 |

| | | | | | | | |
|----|----------------------------------|----------------------------------|----------------------------------|----------------------------------|--------|--------|--------|
| 9 | 87,3016; 16,6667; 49,8118 | 88,2413; 16,6667; 49,4698 | 87,2598; 16,9654; 49,8216 | 88,1944; 17,0262; 49,4711 | 0,0673 | 0,3018 | 3,5259 |
| 10 | 87,3016; 8,3334; 49,8118 | 88,2413; 8,3334; 49,4698 | 86,9603; 9,5617; 50,8778 | 87,9019; 9,5964; 50,5430 | 0,0718 | 1,6618 | 2,0402 |
| 11 | 87,3016; 0,0000; 49,8118 | 88,2413; 0,0000; 49,4698 | 87,9141; 0,0000; 49,8581 | 88,8522; 0,0000; 49,5116 | 0,0284 | 0,6143 | 0,2750 |
| 12 | 87,3016; -8,3334; 49,8118 | 88,2413; -8,3334; 49,4698 | 87,7722; -6,0087; 50,3639 | 88,7121; -6,0439; 50,0243 | 0,0714 | 2,4352 | 2,0230 |
| 13 | 87,3016; -16,6667; 49,8118 | 88,2413; -16,6667; 49,4698 | 86,1170; -15,2148; 52,1840 | 87,0656; -15,2603; 51,8710 | 0,0793 | 3,0230 | 3,1356 |
| 14 | 87,3016; -25,0000; 49,8118 | 88,2413; -25,0000; 49,4698 | 87,1972; -22,4599; 51,0415 | 88,1221; -22,4915; 50,6624 | 0,0788 | 2,8240 | 2,9119 |

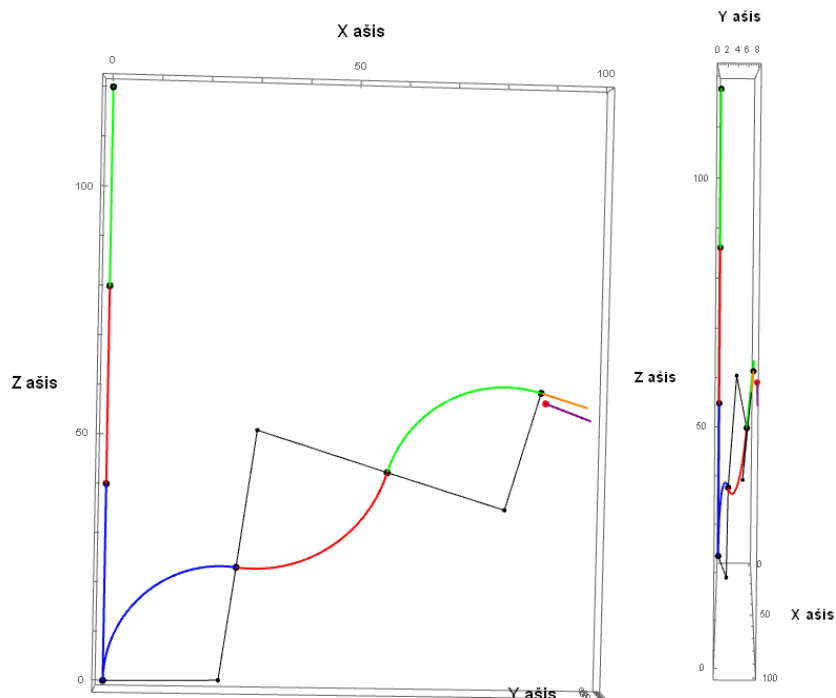
Taikant „Wolfram Mathematica“ programą yra atvaizduojami 38-44 paveikslėliuose roboto išsilankstymai prie 1-7 užduoties taškų pagal 3 ir 4 lentelių duomenis.



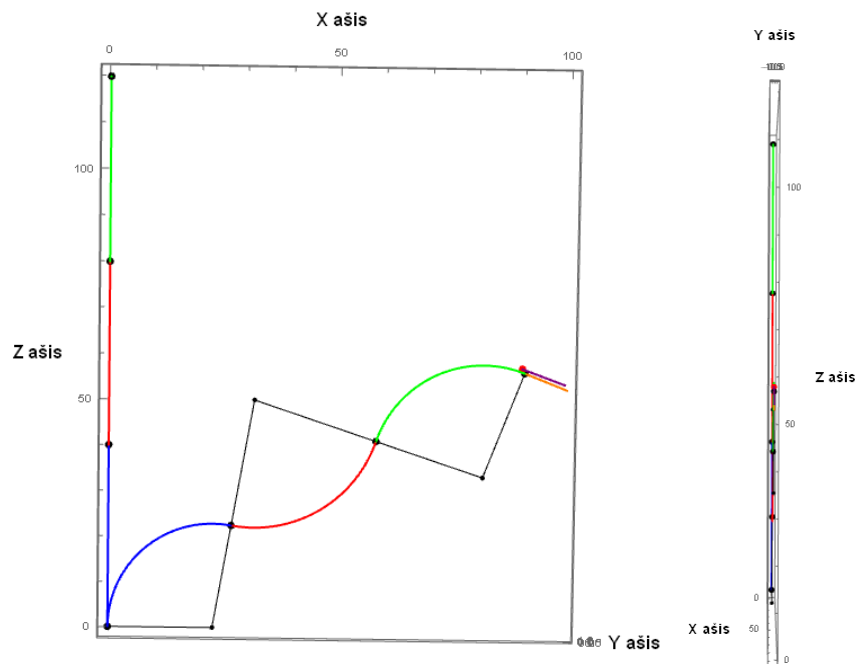
38 pav. Straublio tipo robotas pradinėje padėtyje ir išsilankstęs pagal 3 lentelės duomenis pagal 1-ą užduoties tašką, 1-as rob. segm. – mėlyna spalva, 2-as rob. segm. – raudona spalva, 3-ias rob. segm. – žalia spalva, raudonas taškas esantis prie roboto galo žymi užduoties tašką, o violetinis vektorius žymi iš anksto nurodyto vienetinio vektoriaus kryptį



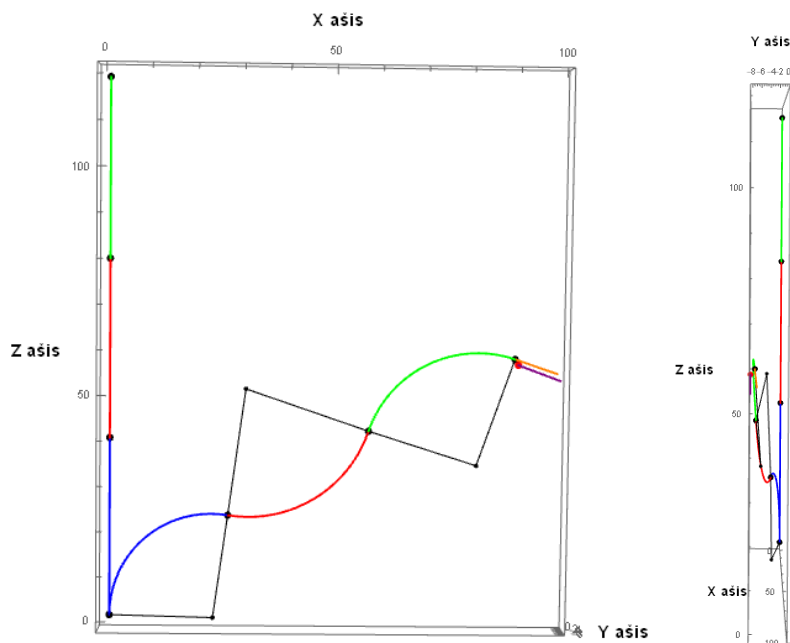
39 pav. Straublio tipo robotas pradinėje padėtyje ir išsilankstęs pagal 3 lentelės duomenis pagal 2-ą užduoties tašką, 1-as rob. segm. – mėlyna spalva, 2-as rob. segm. – raudona spalva, 3-ias rob. segm. – žalia spalva, raudonas taškas esantis prie roboto galo žymi užduoties tašką, o violetinis vektorius žymi iš anksto nurodyto vienetinio vektoriaus kryptį



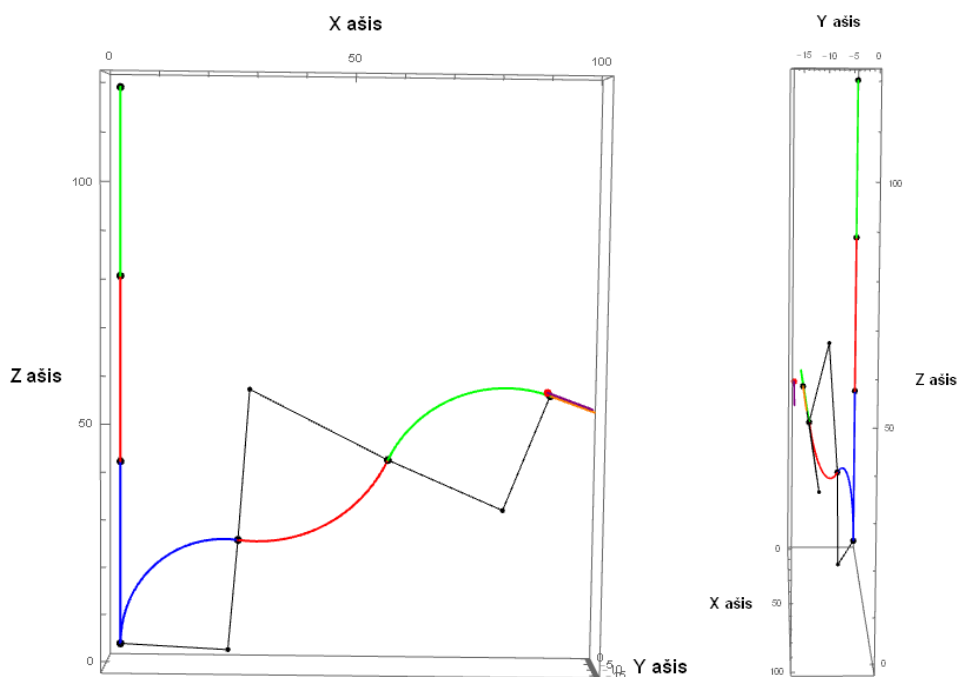
40 pav. Straublio tipo robotas pradinėje padėtyje ir išsilankstęs pagal 3 lentelės duomenis pagal 3-ią užduoties tašką, 1-as rob. segm. – mėlyna spalva, 2-as rob. segm. – raudona spalva, 3-ias rob. segm. – žalia spalva, raudonas taškas esantis prie roboto galo žymi užduoties tašką, o violetinis vektorius žymi iš anksto nurodyto vienetinio vektoriaus kryptį



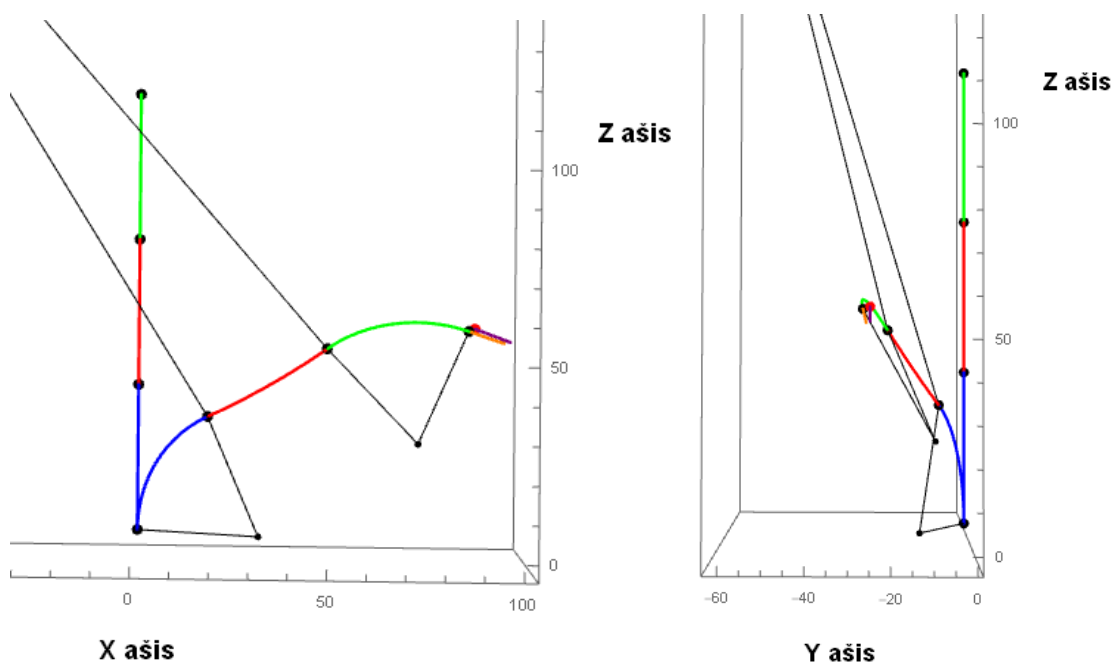
41 pav. Straublio tipo robotas pradinėje padėtyje ir išsilankstęs pagal 3 lentelės duomenis pagal 4-ą užduoties tašką, 1-as rob. segm. – mėlyna spalva, 2-as rob. segm. – raudona spalva, 3-ias rob. segm. – žalia spalva, raudonas taškas esantis prie roboto galo žymi užduoties tašką, o violetinis vektorius žymi iš anksto nurodyto vienetinio vektoriaus kryptį



42 pav. Straublio tipo robotas pradinėje padėtyje ir išsilankstęs pagal 3 lentelės duomenis pagal 5-ą užduoties tašką, 1-as rob. segm. – mėlyna spalva, 2-as rob. segm. – raudona spalva, 3-ias rob. segm. – žalia spalva, raudonas taškas esantis prie roboto galo žymi užduoties tašką, o violetinis vektorius žymi iš anksto nurodyto vienetinio vektoriaus kryptį



43 pav. Straublio tipo robotas pradinėje padėtyje ir išsilankstęs pagal 3 lentelės duomenis pagal 6-ą užduoties tašką, 1-as rob. segm. – mėlyna spalva, 2-as rob. segm. – raudona spalva, 3-ias rob. segm. – žalia spalva, raudonas taškas esantis prie roboto galo žymi užduoties tašką, o violetinis vektorius žymi iš anksto nurodyto vienetinio vektoriaus kryptį



44 pav. Straublio tipo robotas pradinėje padėtyje ir išsilankstęs pagal 3 lentelės duomenis pagal 7-ą užduoties tašką, 1-as rob. segm. – mėlyna spalva, 2-as rob. segm. – raudona spalva, 3-ias rob. segm. – žalia spalva, raudonas taškas esantis prie roboto galo žymi užduoties tašką, o violetinis vektorius žymi iš anksto nurodyto vienetinio vektoriaus kryptį

Poskyrio išvados ir rezultatai:

- Nurodžius 14 užduoties taškų, kuriuos reikia pasiekti 3-jų segmentų straublio tipo roboto galui ir 14 vienetinių roboto galo kryptį rodančių vienetinių vektorių, buvo apskaičiuoti roboto 3-jų segmentų sulenkimo kampai apie y ašis ir pasukimo kampai apie z ašis, kai roboto kiekvieno segmento stuburo ilgis yra lygus 40 cm, o bendras visų trijų segmentų roboto stuburo ilgis yra 120 cm. Vykdam straublio tipo roboto kampų paiešką pagal sukurta roboto valdymo algoritmą buvo nurodyta, jog didžiausias galima bet kurio segmento sulenkimo kampas negali viršyti 120° .
- 14 kartų prie 14-os užduoties taškų išsprendus tiesioginį straublio tipo roboto kinematikos uždavinį pagal sukurtą straublio tipo roboto valdymo algoritmo surastus kampus θ_{g1} , θ_{g2} , θ_{g3} apie y ašis ir kampus ϕ_{g1} , ϕ_{g2} , ϕ_{g3} apie z ašis, buvo gautos išsilanksčiusio straublio tipo roboto galo taško koordinatės ir galo kryptį rodantis vienetinis krypties vektorius kiekvienam roboto išsilankstymo atvejui. Išsprendus tiesioginį straublio tipo kinematikos uždavinį taip pat buvo gautos taškų koordinatės, reikalingos roboto atvaizdavimui „Wolfram Mathematica“ programos aplinkoje.
- Buvo apskaičiuoti straublio tipo roboto euklidiniai atstumai, išreiškiantys roboto pozicionavimo paklaidas ir vektorių krypties nuokrypio kampai, išreiškiantys roboto orientavimo paklaidas. Iš 14 bandymų prie skirtingų užduoties taškų ir krypties vektorių vidutinis euklidinis atstumas lygus apie 1,9248 cm, o vidutinis nuokrypio kampas yra apie $3,2169^\circ$.
- Iš 4 lentelės duomenų matyti, jog šios abi paklaidos yra susiję su surastomis mažiausiomis vektorinių sandaugų ilgių sumomis, kuo pastaroji suma mažesnė, tuo mažesnės pozicionavimo ir orientavimo paklaidos. Norint sumažinti pozicionavimo ir orientavimo paklaidas, vykdam roboto valdymo algoritmą, reikia surasti dar mažesnes vektorinių sandaugų ilgių sumas, kurios beveik būtų lygios 0. Tai galima padaryti tik tada, kai vykdam roboto valdymo algoritmą, 1-o ir 3-io robotų segmentų sulenkimo kampai θ_{g1} , θ_{g3} apie y ašis ir pasukimo kampai ϕ_{g1} , ϕ_{g3} apie z ašis būtų keičiami ne kas 1° , o mažesniu kampu.

Rezultatai ir išvados

1. Išanalizavus straublio tipo konstrukciją, konstrukcinius variantus ir veikimo principą, buvo sukurtas straublio tipo roboto valdymo algoritmas, skirtas robotui, turinčiam 3 pastovaus stuburo ilgio segmentus.
2. Apžvelgti straublio tipo roboto tiesioginės ir atvirkštinės kinematikos uždaviniai ir keli jų sprendimo metodai.
3. Išsprendus straublio tipo roboto trijų segmentų atvirkštinės kinematikos uždavinį dvimatėje koordinatinių sistemoje, pritaikant vektorių savybes, pastebėta, jog bandant apskaičiuoti antro segmento sulenkimo kampą pagal išsireikštą (34) funkciją, patikimesnis būdas yra ją spręsti išskleidus Teiloro eilute. Paėmus, Teiloro eilute išskleisto funkcijos, pirmus 3 narius ir išsprendus kvadratinę lygtį su pakeitimu yra visada randamas atsakymas. Sprendžiant (34) funkciją per Niutono metodą galima rasti tikslesnį atsakymą, tačiau pastarasis metodas turi du didelius trūkumus, tai yra nepastovų iteracijų skaičių ir savybę diverguoti, dėl ko yra nerandamas ieškomas atsakymas.
4. Ištyrus sukurto straublio tipo roboto valdymo algoritmo greitaveiką, buvo nustatyta, jog sukurtos programos algoritmas yra gana lėtas (vidutiniškai vienam roboto išsilankstymui reikia rasti 4 minučių 24 sekundžių), tačiau skaičiavimo trukmę galima sumažinti pritaikius modifikuotą nuoseklios paieškos metodą su Dichotomijos metodo idėja, kampui θ_{g1} parinkti (skaičiavimo trukmė sumažėjo iki 1 minutės 58 sekundžių).
5. Ištyrus sukurto straublio tipo roboto valdymo algoritmo pozicionavimo tikslumą nustatyta, jog surasti roboto išsilankstymai nurodyto užduoties taško ir orientacijos tiksliai nepasiekia, (vid. euklidinis atstumas yra apie 1,9248 cm, o vid. nuokrypio kampas yra 3,2169 °). Siekiant didesnio pozicionavimo tikslumo reikia 1 ir 3 robotų segmentus lenkti θ_{g1} , θ_{g3} apie y ašis ir sukti ϕ_{g1} , ϕ_{g3} apie z ašis mažesniais negu 1° kampais.

Literatūros sąrašas

1. D. B. Camarillo, C. F. Milne, C. R. Carlson, M. R. Zinn and J. K. Salisbury, Mechanics modeling of tendon-driven continuum manipulators, *IEEE Trans. Robotics* 24, 1262–1273, (2008). Prieiga per internetą: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4660316>
2. R. J. Webster III, J. M. Romano and N. J. Cowan, Mechanics of precurved-tube continuum robots, *IEEE Trans. Robotics* 25, 67–78, (2009). Prieiga per internetą: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.691.2052&rep=rep1&type=pdf>
3. Rongjie Kang^{1,2}, David T. Branson³, Tianjiang Zheng¹, Emanuele Guglielmino¹ and Darwin G. Caldwell. Design, modeling and control of a pneumatically actuated manipulator inspired by biological continuum structures, (2012).
4. Straublio tipo roboto segmento sulenkimo kampai. Prieiga per internetą: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5806630/>
5. Roborų segmento laisvės laipsniai ir „sausgyslių galimi išdėstymai“. Prieiga per internetą: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4819480/>
6. Straublio tipo roboto tipai, kinematika, veikimo principas. Prieiga per internetą: <https://www.hindawi.com/journals/isrn/2013/726506/>
7. Šepetėlinių ir bešepetėlinių nuolatinės srovės variklių privalumai ir trūkumai. Prieiga per internetą: <https://www.dpairless.com/blog/cons-and-pros-of-motorbrushless-and-brushed-motor.html>
8. Kintamos srovės sinchroninio variklio su nuolatiniais magnetais efektyvumo kreivės. Prieiga per internetą: <https://empoweringpumps.com/ac-induction-motors-versus-permanent-magnet-synchronous-motors-fuji/>
9. BLDC variklių konstrukcija ir jos tipai, veikimo principas, klasikiniai ir modernūs variklio valdymo metodai. Prieiga per internetą: <https://dergipark.org.tr/download/article-file/256949>
10. Oro tarpas tarp statoriaus ir rotoriaus BLDC varikliuose. Prieiga per internetą: <https://www.avrfreaks.net/forum/15-cm-distance-between-stator-and-rotor>
11. BLDC variklio struktūra kai magnetai rotoriaus išorėje arba viduje. Prieiga per internetą: <https://www.controleng.com/articles/understanding-permanent-magnet-motors/>
12. Elektros variklio apibrėžimas ir jų tipai. Prieiga per internetą: <https://circuitglobe.com/electric-motor.html>
13. BLDC variklio valdymas ir jo metodai. Prieiga per internetą: <https://www.renexas.com/eu/en/solutions/key-technology/motor-control/motor-algorithms/bldc.html>
14. BLDC variklio sinusinis valdymas. Prieiga per internetą: <https://www.nxp.com/docs/en/application-note/AN4869.pdf>
15. BLDC variklio FOC valdymas. Prieiga per internetą: <https://www.edn.com/Home/PrintView?contentItemId=4439137>
16. Azamat Yeshmukhametov, Koichi Koganezawa, Yoshio Yamamoto. Developing a Kinematically Similar Master Device for Extensible Continuum Robot Manipulators, (2019).
17. Straublio tipo robotas, kurio segmentas yra disko-guolio-disko tipo. Prieiga per internetą: <https://www.hindawi.com/journals/isrn/2013/890609/>

18. Azamat Yeshmukhametov, Zholdas Buribayev, Yedilkhan Amirgaliyev, Beibut Amirgaliyev, Konstantin Latuta. Bio-Inspired A Novel Continuum Robot Arm With Variable Backbone Design: Modelling And Validation, (2019).
19. Thien-Dang Nguyen and Jessica Burgner-Kahrs. A Tendon-Driven Continuum Robot with Extensible Sections, (2015).
20. Roboto manipulatoriaus Denavito-Hartenbergo parametrų. Prieiga per internetą:
<http://blog.robindeits.com/2012/06/10/denavit-hartenberg-for-robotics-part-3-the-d-h-parameters/>
21. Srinivas Neppalli, Bryan A. Jones. Design, Construction, and Analysis of a Continuum Robot (2007).
22. Oilerio kampai ir posūkių matricos. Knyga: „Robotų technika“, autor. Bronius Bakšys, Algimantas Fedaravičius, 2005m.
23. 12 Oilerio kampų sistemų. Prieiga per internetą:
<https://rotations.berkeley.edu/the-euler-angle-parameterization/>
24. Bryan A. Jones, Ian D. Walker. Kinematics for Multisection Continuum Robots, (2006).
25. Arnau Garriga-Casanovas, Ferdinando Rodriguez y Baena. Kinematics of Continuum Robots With Constant Curvature Bending and Extension Capabilities, (2019).
26. Srinivas Neppalli, Matthew A. Csencsits, Bryan A. Jones, and Ian Walker. A Geometrical Approach to Inverse Kinematics for Continuum Manipulators, (2008).
27. Richard Sperling. Design Of A Tendon Driven Soft Continuum Robotic Arm Using Proprioceptive Sensors, (2018).
28. Guangzhu Meng, Guangming Yuan, Zhe Liu, Jun Zhang. Forward and inverse kinematic of continuum robot for search and rescue, (2013).
29. Niutono skaičiavimo metodų funkcijos kintamajam rasti. Prieiga per internetą:
<http://mathcentral.uregina.ca/qq/database/qq.09.00/roble1.html>
30. Funkcijos skleidimas Teiloro eilute. Prieiga per internetą:
<https://www.physicsforums.com/threads/simple-how-do-you-solve-sin-x-x-0-99.150413/>
31. Dichotomijos paieškos metodai. ANTONIOU, A., Wu-Sheng Lu, Practical Optimization, Algorithms and Engineering Applications, Canada: Springer, (2007).