

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

Egidijus Pakėnas

**PASLAUGŲ ROBOTO, JUDANČIO SPIRALINIAIS LAIPTAIS,
VALDYMO TYRIMAS**

Baigiamasis magistro projektas

Vadovas

Prof. dr. Vytenis Sinkevičius

PANEVĖŽYS, 2017

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

**PASLAUGŲ ROBOTO, JUDANČIO SPIRALINIAIS LAIPTAIS,
VALDYMO TYRIMAS**

Baigiamasis magistro projektas
Valdymo technologijos (kodas 621H66001)

Vadovas

Prof. dr. Vytenis Sinkevičius

Recenzentas

Projektą atliko

Egidijus Pakėnas

PANEVĖŽYS, 2017

2017-__-__

BAIGIAMOJO PROJEKTO UŽDUOTIS

Išduota studentui: Egidijui Pakėnui Grupė PME-5

1. Darbo tema:

Lietuvių kalba: Paslaugų roboto, judančio spiraliniais laiptais, valdymo tyrimas

Anglų kalba: Research on Control of Service Robot Moving by Spiral Staircases

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

2. Darbo tikslas:

Sudaryti paslaugų roboto, judančio spiraliniais laiptais, modelį ir ištirti valdymo galimybes, reikalingam judesiui realizuoti.

3. Reikalavimai ir sąlygos:

Ištirti mobilias platformas, ištirti pasirinktą platformą ir jai sukurti valdymo sistemą.

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

1. *Literatūros analizė;*
2. *Mobilios platformos judėjimo spiraliniais laiptais analizė;*
3. *Valdymo sistemos, skirtos judėjimui spiraliniais laiptais, analizė;*
4. *Gautų rezultatų apibendrinimas.*

5. Ši užduotis yra neatskiriama baigiamojo projekto dalis.

6. Projekto pateikimo gynimui kvalifikacinėje komisijoje terminas

2017-06-01

(data)

Užduotį gavau: Egidijus Pakėnas

(studento vardas, pavardė, parašas)

2017-02-01

(data)

Vadovas: Prof. Vytenis Sinkevičius

(pareigos, vardas, pavardė, parašas)

2017-02-01

(data)



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Panevėžio technologijų ir verslo

(Fakultetas)

Egidijus Pakėnas

(Studento vardas, pavardė)

Valdymo technologijos 621H66001

(Studijų programos pavadinimas, kodas)

„Baigiamojo projekto pavadinimas“
AKADEMINIO SAŽININGUMO DEKLARACIJA

20 17 m. gegužės 31 d.
Panevėžys

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

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

(vardą ir pavardę įrašyti ranka)

(parašas)

Pakėnas, Egidijus. Paslaugų roboto, judančio spiraliniais laiptais, valdymo tyrimas. Magistro baigiamasis projektas / vadovas prof. Vytenis Sinkevičius; Kauno technologijos universitetas, Panevėžio technologijų ir verslo fakultetas.

Mokslo kryptis ir sritis: Valdymo technologijos, 621H66001

Reikšminiai žodžiai: *mobili platforma, Rocker-Bogie, greičio valdymas, spiraliniai laiptai.*

Panevėžys, 2017. 61 p.

SANTRAUKA

Magistro baigiamajame darbe tiriamos mobilios platformos, galimos panaudoti kaip paslaugų robotai, skirti perkelti krovinius ar žmones spiraliniais laiptais. Pasirinktos mobilios platformos judėjimas spiraliniais laiptais išanalizuotas matematiškai, pagal gautą modelį sukurta ir ištirta valdymo sistema.

Mokslinės literatūros analizėje išanalizuotos sukurtos ir kuriamos mobilios platformos, jų judėjimo principai. Iš ištirtų platformų pasirinkta viena, skirta tolimesnei analizei. Mobilus platforma pasirinkta pagal konstrukcinį paprastumą ir geriausias kliūčių įveikimo savybes.

Matematinėje platformos judėjimo analizėje išanalizuotas platformos judėjimas spiraliniais laiptais. Judėjimui aprašyti naudojamos platformos ratų centrų koordinatės, reikalingos valdymo sistemos analizei.

Trečiojoje projekto dalyje sukuriama algoritmas mobilios platformos valdymo sistemai ir algoritmas, skirtas platformos judėjimo simuliacijos programai „*MATLAB*“ programiniame pakete. Atlikus simuliaciją, įvertinama ištirta valdymo sistema ir aprašomos techninės galimybės jai realizuoti.

Darbo pabaigoje pateiktos išvados, naudotos literatūros sąrašas, priedai.

Pakėnas, Egidijus. *Research on Control of Service Robot Moving by Spiral Staircases*: Master thesis in Master of Engineering Sciences / supervisor assoc. prof. Vytenis Sinkevičius. Panevėžys Faculty of Technologies and Business, Kaunas University of Technology.

Research area and field: Control Technologies, 621H66001

Key words: *mobile robot platform, Rocker-Bogie, speed control, spiral staircase*

Panevėžys, 2017. 61 p.

SUMMARY

In this master project mobile robot platforms were investigated in order to use them as stair-climbing service robots for spiral staircases. The mathematical model of selected mobile robot platform movement was created and was used for further investigation of its control system.

The first part of the research investigates mobile robot platforms that are developed or being developed. Using structural simplicity and ability to overpass obstacles as criteria, the best type of mobile robot platform was selected for further investigation.

Mathematical model investigates the wheel center position of each wheel while climbing spiral staircases. The center position of each wheel is needed for investigating the control system of selected mobile robot platform.

In the third part of the research, algorithms for mobile robot platform control system and its simulation in „MATLAB“ are created. Using the results of simulation, the functional model of technical control system is created.

In the end of the project, conclusions, references and attachments are listed and given.

TURINYS

IVADAS	9
1. MOKSLINĖS LITERATŪROS APŽVALGA	10
1.1. Žvaigždės tipo ratų modulių turinčios platformos	10
1.1.1. Neįgaliojo vežimėlis, kuriame yra naudojami elektros varikliai	11
1.1.2. Platforma, kurioje naudojami hidrauliniai varikliai	13
1.2. „RHEX“ tipo lipančios platformos.....	15
1.3. „Rocker-Bogie“ tipo platformos.....	19
1.3.1. Platformos mechanika ir kinematika	19
1.3.2. Galima valdymo sistema	21
1.4. Skyriaus išvados	23
2. MATEMATINĖ MOBILIOS PLATFORMOS ANALIZĖ	24
2.1. Sistemos aprašymas	24
2.2. Pirmojo ratuko judėjimas laiptais	25
2.3. Antrojo ratuko judėjimas laiptais	27
2.3.1. Antrojo ratuko judėjimas horizontalia plokštuma	27
2.3.2. Antrojo ratuko judėjimas vertikalia plokštuma	28
2.3.3. Antrojo ratuko vertinimas per išorinę laiptelio briauną.....	30
2.4. Trečiojo ratuko judėjimas laiptais	31
2.4.1. Šarnyro H koordinatinių radimas	31
2.4.2. Trečiojo ratuko judėjimas horizontalia plokštuma	32
2.4.3. Trečiojo ratuko judėjimas vertikalia plokštuma	33
2.4.4. Trečiojo ratuko vertinimas per išorinę laiptelio briauną.....	35
2.5. Platformos judėjimo netolygumas	36
2.6. Skyriaus išvados	37
3. PLATFORMOS VALDYMAS PAGAL LAIPTŲ FORMĄ	38
3.1. Programos algoritmas	38
3.2. Spiralinių laiptų aprašymas	39
3.3. Įvestinių duomenų nuskaitymas	40
3.4. Judėjimas horizontaliai	41
3.5. Judėjimas vertikaliai	42
3.6. Ratuko vertinimas per išorinę laiptelio briauną	43
3.7. Simuliacijos rezultatai	44
3.8. Techninė valdymo sistema	46
3.9. Platformos valdymo algoritmas.....	47

3.10. Ratų posūkio kampų valdymas pagal ratų ašių atstumus iki laiptų.....	47
3.11. Skyriaus išvados	49
IŠVADOS	50
LITERATŪROS SĄRAŠAS.....	51
PRIEDAI	52
1 PRIEDAS. LAIPTŲ MODELIS „ <i>MATLAB</i> “ PROGRAMINĖJE APLINKOJE.....	53
2 PRIEDAS. PLATFORMOS APRAŠYMAS IR JUDĖJIMAS LAIPTAIS	55
3 PRIEDAS. A PUSĖS RATUKŲ JUDĖJIMAS SPIRALINIAIS LAIPTAIS	57

ĮVADAS

Judėjimo problema žmonijai ir gyvūnijai yra aktuali nuo pat jos atsiradimo pradžios, tačiau kiekvienas sveikas žmogus ar kitas gyvas organizmas yra prisitaikęs judėti įvairiomis sąlygomis ir paviršiais. Negalintiems judėti, jau sukurta įvairių sprendimų ir nuolat ieškoma naujų.

Sparčiai tobulėjant technologijoms, automatizuojant bei robotizuojant įvairius procesus, ieškoma modernių sprendimų platformų judėjimui nelygiais paviršiais. Platformos per kliūtis turi pervežti įvairius krovinius, perkelti žmones, todėl kliūčių įveikimas – pavojingas ir atsakingas darbas. Kuriant šiam tikslui skirtas platformas, svarbu atsižvelgti į galimus pavojus žmogui ir kroviniui, su kuriais jis gali susidurti transportavimo metu, išanalizuoti paslaugų robotų judėjimo bei valdymo principus.

Darbe siekiama išanalizuoti mobilią platformą, skirtą kliūčių įveikimui. Didžiausias dėmesys skiriamas platformų judėjimui laiptais aukštyn, kadangi tai yra sunkiausias uždavinys judėjimo sutrikimui turinčiam žmogui. Nagrinėjamos sukurtos ar kuriamos platformos ir jų prototipai, aiškinami jų judėjimo laiptais principai. Pasirenkama viena iš ištirtų platformų ir tiriamas jos judėjimas spiraliniais laiptais pagal sukurtą valdymo sistemą.

Tyrimo objektas – laiptais judanti platforma.

Darbo tikslas – sudaryti paslaugų roboto, judančio spiraliniais laiptais, modelį ir ištirti valdymo galimybes, reikalingam judesiui realizuoti.

Darbo uždaviniai

1. Išanalizuoti mobilią platformą, kurios gali judėti laiptais;
2. Išanalizuoti pasirinktos mobilios platformos judesį ir valdymą, platformai judant spiraliniais laiptais.

Tyrimo metodai

1. Mokslinės literatūros analizė;
2. Matematinė platformos analizė;
3. Kompiuterinė platformos analizė.

1. MOKSLINĖS LITERATŪROS APŽVALGA

Skyriuje apžvelgiami mobilių platformų tyrimai, koncepcijos ir judėjimo principai. Didžiausias dėmesys skiriamas judėjimui laiptais, kadangi pagal judėjimo savybes bus pasirenkama tinkamiausia sistema tolesniems tyrimams.

1.1. Žvaigždės tipo ratų modulių turinčios platformos

Daugiaratės sistemos jau ilgą laiką yra naudojamos įvairiose keltuvuose ir vežimėliuose. Šios sistemos yra pakankamai stabilios, paprastos ir palengvina įvairių kliūčių įveikimą ar lipimą laiptais. Nors žvaigždės tipo ratų moduliai yra plačiai naudojami ir pritaikomi, tačiau jie turi gana rimtą problemą – tokių ratų sistemos gali judėti tik tiesia trajektorija. Tai reiškia, kad įveikti kliūtį posūkyje ar kilti bei leistis spiraliniais laiptais autonominės sistemos negali arba jų judėjimas bei valdymas tampa labai komplikuoti.



1 pav. Rankinis vežimėlis su žvaigždinių ratų moduliais [1]

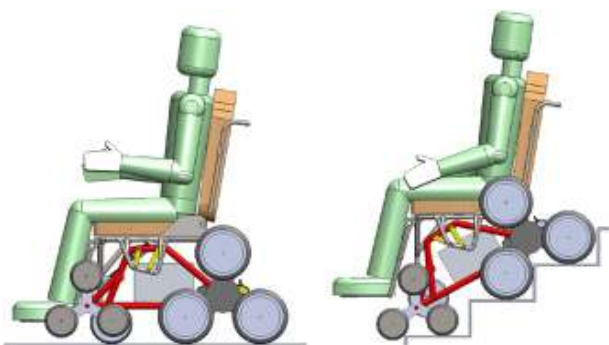


2 pav. Neįgaliojo vežimėlis su žvaigždinių ratų moduliais [2]

1.1.1. Neįgaliojo vežimėlis, kuriame yra naudojami elektros varikliai

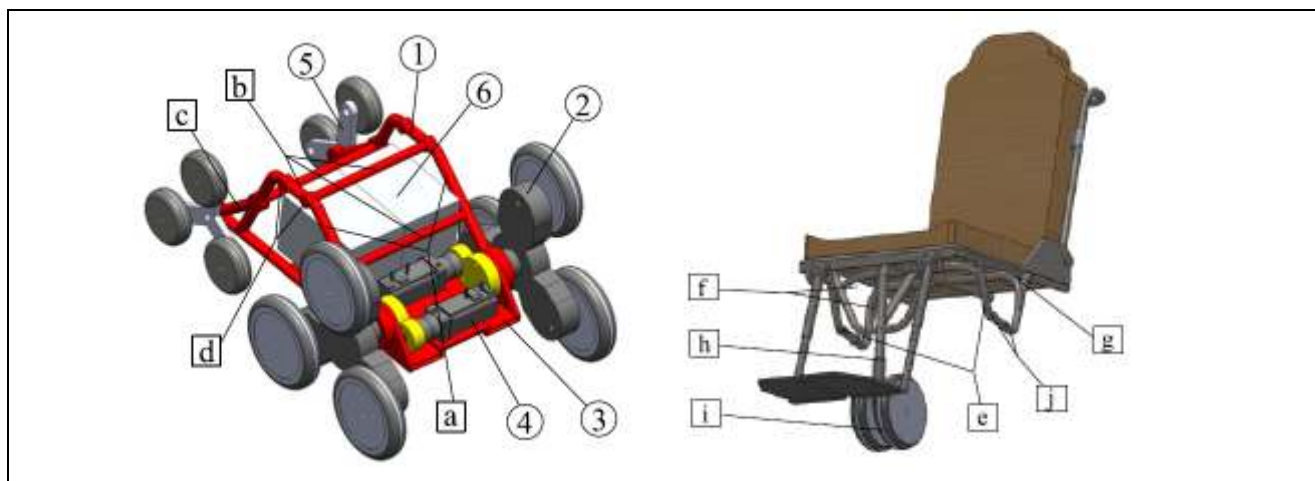
Poskyryje tiriamas neįgaliojo vežimėlio prototipas, turintis elektros varikliais varomus žvaigždinių ratų modulius [3].

Šiame poskyryje nagrinėjama vežimėlio koncepciją sudaro trys pagrindinės dalys: rėmas, kėdė ir ją laikanti konstrukcija, elektrinės pavaros. Kuriant prototipą buvo laikomasi idėjos, kad sistema turi būti nesudėtinga techniniu požiūriu (judesys sukuriamas dviejų elektros variklių ir perduodamas naudojanti dvi transmisijos sistemas), o pats vežimėlis pasyviai keistų judėjimo tipą iš judėjimo lygiu paviršiumi į jos įveikimą. Pasyvus elektrinio vežimėlio lipimas laiptais žemyn pavaizduotas žemiau esančiame paveiksle (3 pav.).



3 pav. Leidimasis laiptais naudojant pasyvaus valdymo vežimėlį [3]

Vežimėlį sudaro šie elementai: vežimėlio rėmas (1), dvi mechaninės pavaros (2), rėmas (3) dvi elektromechaninės pavaros (4), du žvaigždės tipo trijų ratų moduliai (5) ir akumulatoriai (6) (4 pav.).

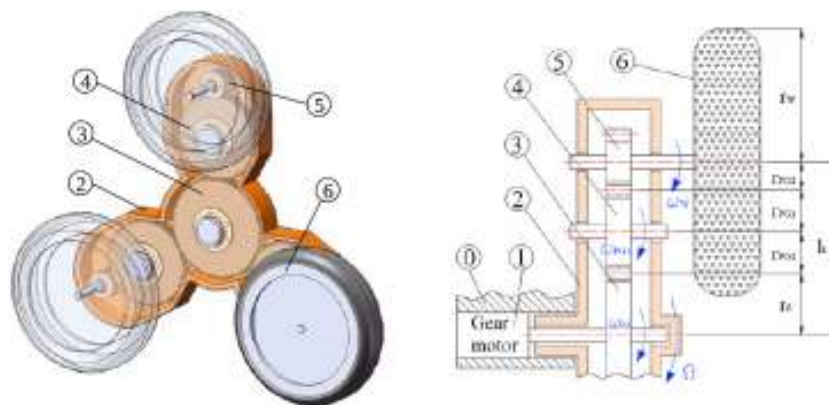


4 pav. Vežimėlio konstrukcija [3]

Kadangi elektrinis neįgaliojo vežimėlis yra skirtas kliūčių įveikimui bei lipimui laiptais tiek aukštyn, tiek žemyn, todėl labai svarbu užtikrinti kuo didesnę žmogaus saugumą ir vežimėlio stabilumą. Šiai problemai išspręsti naudojamas atraminis kėdės ratas (i), kuris lipimo laiptais metu visada turi kontaktą su paviršiumi. Šis sprendimas pašalina nestabilumą ir galimą vežimėlio paslydimą lipant laiptais žemyn. Mažųjų ratų skersmuo parinktas atsižvelgiant į tai, kad jie skirti tik

lipimui laiptais bei vežimėlio atramai, o galiniai varomieji ratai turi būti žymiai didesni už priekinius. Simuliacijų metu nustatyta, kad tokia vežimėlio konstrukcija leidžia įveikti ne aukštesnes kaip 21cm aukščio kliūtis. Tam, kad vežimėlis saugiai ir sklandžiai liptų laiptais, reikalinga pakankamai didelė ratų bazė, svorio centras turi būti kiek galima žemiau, o vežimėlio konstrukcija turi užtikrinti stabilų ir saugų naudojimąsi juo.

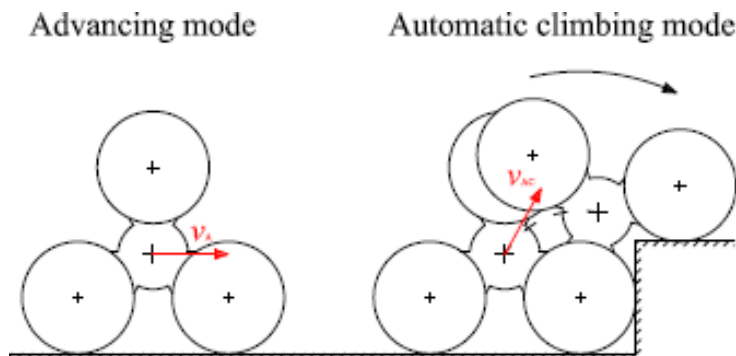
Vežimėlio varančioji ašis yra galinė, o priekiniai ratų moduliai yra mechaniškai bei elektriškai nevaromi. Kiekvienas iš priekinių ratų yra pritvirtintas prie žvaigždės tipo laikiklių, kurie vienas kito atžvilgiu yra pasukti 120 laipsnių. Kiekvienas mažasis ratas gali sukis apie savo ašį, o visi kartu – apie bendrą modulio ašį. Priekinių ratų judesys yra laisvas, o galiniai ratai varomi dviejų elektrinių variklių. Galiniai varantieji vežimėlio ratų moduliai projektuojami taip, kad kiekvieną modulį valdytų tik vienas elektros variklis ir epiciklinė krumpliaračių sistema su keletu laisvės laipsnių skaičių, užtikrinančių skirtingas vežimėlio judėjimo galimybes. Ratų modulis yra sudarytas iš elektros variklio (1), korpuso, kuriame yra įmontuotas krumpliaračių mechanizmas (2), didžiojo krumpliaračio, nuosekliai prijungto prie elektros variklio ir sukančio visus modulio ratus vienu metu (3), pirminio krumpliaračio (4), antrinio krumpliaračio (5) ir pačio rato (6) (5 pav.). Turint tokią elektromechaninę pavarą, galima pasyviai keisti vežimėlio judėjimo režimą iš važiavimo ant ratų ar lipimo laiptais (per kliūtį). Šios sistemos pagrindinis trūkumas – nuolat sukami visi ratai, todėl ne visa sunaudojama elektros energija virsta į reikalingą mechaninį judesį, o krumpliaračių sistema papildomai dėvisi. Varantieji ratų moduliai yra tarpusavyje nesujungti, tačiau įmontuoti į bendrą rėmą (0) ir gali laisvai sukis.



5 pav. Varančiojo ratų modulio struktūra [3]

Pasyvaus valdymo elektrinis vežimėlis gali judėti slenkamuuju režimu (angl. advancing mode) arba įveikti atsiradusias kliūtis (angl. automatic climbing mode). Vežimėlio judėjimo režimas priklauso nuo to, kokį pasipriešinimo momentą reikia nugalėti elektrinėms pavaroms ir priekiniams ratų moduliams. Kuomet vežimėlis juda tolygiu paviršiumi, žvaigždinius ratų laikiklius nuolat veikia vežimėlio sunkio jėga ir elektros variklių sukuriamas sukimo momentas, todėl prie lygaus paviršiaus prispaudžiami kiekvieno modulio du ratai. Važiuojant tokiu režimu, vežimėlį iš

viso varo keturi ratai. Kai privažiuojama kliūtis ir ratai į ją atsiremia, vežimėlį varančioji jėga prispaudžia priekinį ratą prie kliūties taip, kad žvaigždinis ratų laikiklis verčiasi apie savo ašį, nes pasikeičia judėjimo greičio vektoriaus kryptis ir viršutinis ratas užlipa ant kliūties – taip gaunamas lipimo per kliūtį judesys. Žemiau esančiame paveikslėlyje (6 pav.) pavaizduotas lipimas laiptais aukštyn.



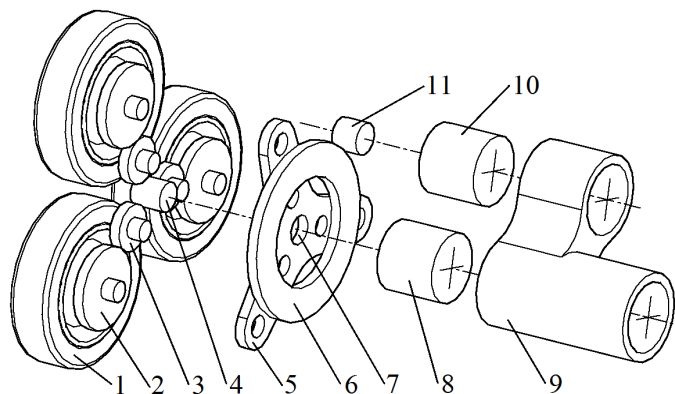
6 pav. Judėjimo režimo keitimasis [3]

Ištirtas elektrinio vežimėlio prototipas kelia minimalius elektros pavarų valdymo reikalavimus – tereikia valdyti elektros variklius juos įjungiant ar išjungiant. Tai sumažina vežimėlio valdymo sukuriamas galimas problemas, kadangi judesio režimo keitimas pagrįstas tik dinaminėmis, o ne elektriškai kontroliuojamomis judėjimo savybėmis.

1.1.2. Platforma, kurioje naudojami hidrauliniai varikliai

Šiame poskyryje nagrinėjama panašios struktūros platforma [4], aprašyta 1.1 poskyryje, kuriai judėjimas tolygiu bei laiptuotu paviršiumi yra sukuriamas hidraulinių variklių. Šios platformos pavaros yra sudėtingesnės, vienam ratų moduliui naudojamas ne vienas, o du varikliai, kurie dirba ne vienu metu, o tik vienas vienu metu, priklausomai nuo reikalingo judėjimo tipo.

Platformos ašį sudaro keturi identiški trijų ratų moduliai, (7 pav.), kurie nepriklausomai vienas nuo kito yra sukami hidraulinių variklių.



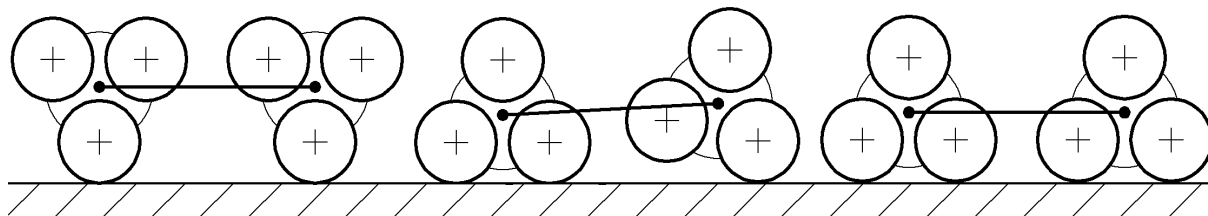
7 pav. Hidraulinė vežimėlio pvara [4]

Ratai (1) yra prijungti prie žvaigždės tipo disko (5), prie kurio yra standžiai pritvirtintas dydysis krumpliaratis (6). Priklausomai nuo reikiamo judėjimo tipo (slenkamasis arba lipimas laiptuotu paviršiumi), ratus suka hidraulinis variklis 1 (8) arba hidraulinis variklis 2 (10), kurie yra įmontuoti bendrame korpuse (9).

Kuomet reikalingas slenkamasis judėjimas lygiu paviršiumi, ratus suka hidraulinis variklis 1. Šio variklio ašis yra nuosekliai prijungta prie krumpliarachio (4), kuris taip pat yra žvaigždinio disko ašis (7). Kuomet šis krumpliaratis sukasi, jis taip pat per mažesnius krumpliarachius (2 ir 3) suka visus modulio ratus kartu – žvaigždinis diskas nesisuka apie savo ašį, todėl gaunamas tolygus judėjimas, o ratų modulis su paviršiumi liečiasi dviem ratais.

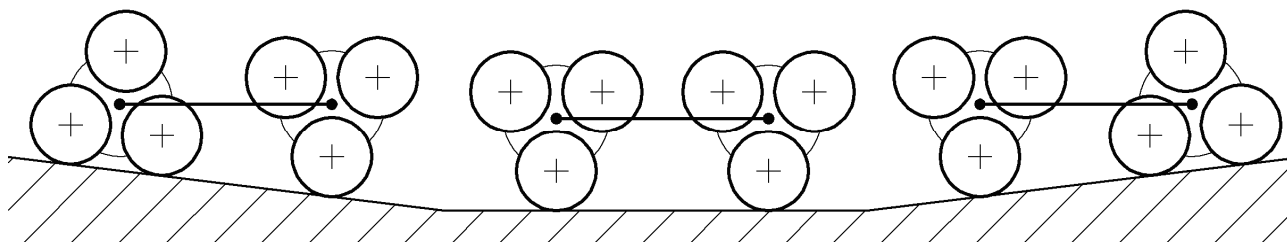
Lipant laiptuotu paviršiumi ar įveikiant kliūtį, hidraulinis variklis 1 yra nenaudojamas, o naudojamas hidraulinis variklis 2. Šis variklis suka krumpliaratį 11, kuris per dydyjį krumpliaratį suka žvaigždinį diską, o kartu ir visus tris modulio ratus aplink bendrą jų ašį – taip gaunamas platformos lipimas laiptais.

Kadangi ratų modulis yra valdomas dviejų atskirų variklių, todėl galimos įvairios kombinacijos, kurios apibūdina kurie ratai liečia paviršių platformai judant (8 pav.). Iš paveikslu matyti, kad geriausias judesys gaunamas tuomet, kai vienas ratų modulis paviršių liečia dviem ratais – dėl žemo masės centro gaunamas didžiausias stabilumas, maksimaliais panaudojamas variklio sukuriamas sukimo momentas, platformos pagrindas išlaikomas idealioje horizontalioje padėtyje.



8 pav. Platformos judėjimo lygiu paviršiumi galimybės [4]

Aktyvus ratų modulių valdymas leidžia ratus pasukti reikiamu kampu taip, kad būtų gautas reikiamas judėjimo tipas – žemiau pateiktu atveju, kai posvyris yra 7° (9 pav.), horizontalus pagrindo išlaikymas. Esant horizontalioje padėtyje, pakeliamas platformos masės centras ratų modulius pasukant 120° kampu (vidurinė padėtis). Po to, priklausomai nuo krypties, į kurią juda vežimėlis (aukštyn ar žemyn), atitinkamai priekinės arba galinės ašies svorio centras yra leidžiamas žemyn, išlaikant pagrindą horizontalioje padėtyje.



9 pav. Horizontalios padėties išlaikymas [4]

Hidrauliniai varikliai sukuria didesnę sukimo momentą, todėl yra patogūs naudoti platformose, kurios turi perkelti didelės masės krovinius. Tačiau tokia sistema yra sudėtingesnė, lyginant su tokia, kurioje naudojami elektros varikliai, sudėtingesnis techninis aptarnavimas.

1.2. „RHEX“ tipo lipančios platformos

„RHex“ – tai robotas, kurio judesys yra pagrįstas vabzdžių judėjimo principu. Pagrindinė paskirtis – judėti nelygiu paviršiumi ir įveikti kliūtis, kurios gali būti net iki keleto kartų didesnės už pačius robotus. Lyginant su ratinėmis platformomis, kojas turinčios platformos pasižymi geresnėmis kliūčių įveikimo savybėmis, tačiau turi mažesnį korpuso stabilumą. Korpuso stabilumas yra itin svarbus, jeigu platforma bus naudojama lengvai deformuojantiems kroviniams ar žmonėms gabenti. Skyriuje tiriamos „RHex“ roboto dinaminės judėjimo ir reikalingo valdymo tokiam judesiui realizuoti galimybės [5]

Šie robotai yra autonominiai ir gali įveikti daugumą gamtoje esančių kliūčių. Visi „RHex“ tipo platformų vykdymo įtaisai yra roboto konstrukcinė dalis. Svarbiausias elementas – roboto kojos. Šios kojos turi tik vieną laisvės laipsnį, tačiau yra esminis konstrukcinis elementas. Palyginus su bendroju roboto svoriu, kojos sudaro tik labai mažą jo dalį.



10 pav. RHex tipo robotas [6]

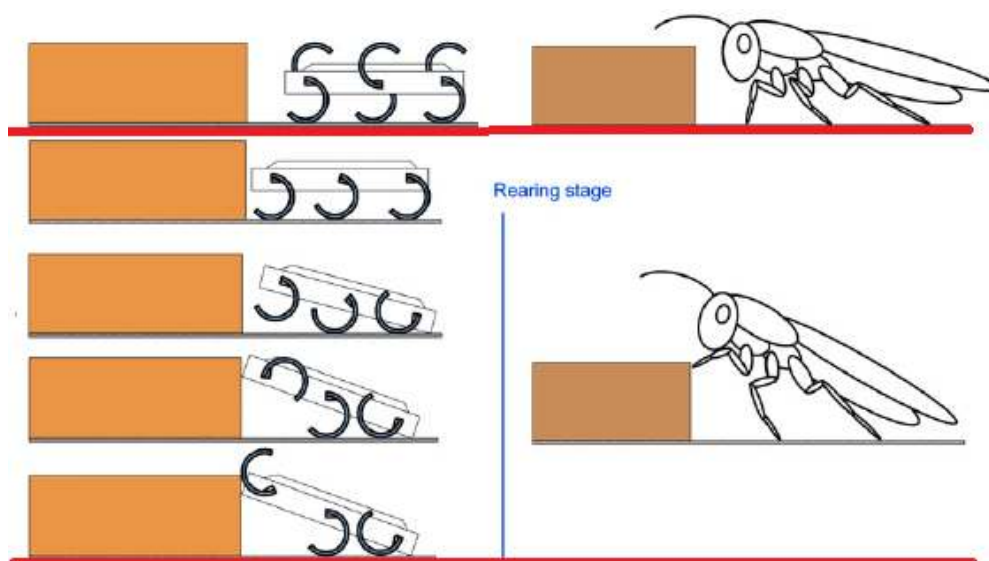


11 pav. RHex robotas lipa per kliūtį [7]

Nagrinėjamas robotas sukurtas remiantis tarakono judėjimo principu ir nuo įprastinių laiptais lipančių platformų skiriasi tuo, kad naudojami ne įprastiniai ratai, o pusbėnelio formos kojos. Platforma gali turėti keturias arba šešias kojas. Skirtingai nuo ratinių robotų, kojas turinčių robotų tyrimai koncentruoti labiau į kojų valdymą, o ne patį kliūties aptikimą, todėl šiame poskyryje nagrinėjamas tik „RHex“ tipo roboto judėjimo principas.

Tarakono kliūties įveikimą galima padalinti į du žingsnius – pritūpimo (angl. *rearing*) ir kylimo (angl. *rising*). Pritūpimo metu tarakonas pritūpia galinėmis kojomis, taip savo galinę kūno dalį nuleisdamas prie paviršiaus, ant kurio yra, o priekinę dalį pakeldamas aukštyn. Pritūpimas vyksta prieš tai, kol tarakonas padeda bent vieną koją ant įveikiamos kliūties. Kylimo metu tarakonas kelia savo masės centrą aukštyn taip, kad jo kūno posvyrio kampas nekistų arba kistų kaip galima mažiau.

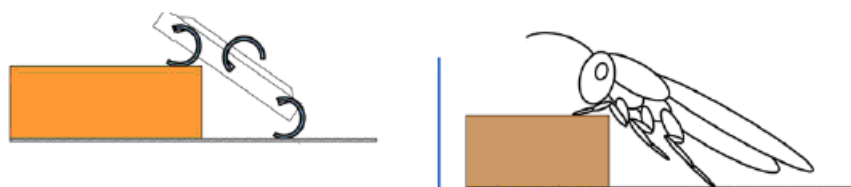
Panašiu principu laiptais gali lipti ir šiame skyriuje nagrinėjamas robotas, o patį procesą galima padalinti į septynis žingsnius. Platformos persikėlimas per kliūtį priklauso nuo, kaip sėkmingai ir tolygiai pusbėnelio formos kojos gali perkelti roboto masės centrą per kliūtį. Roboto ir tarakono lipimo panašumai pavaizduoti žemiau esančiuose paveiksluose (12-17 pav.); žingsniai atskirti horizontalia stora linija. Kiekvienam kliūties tipui įveikti naudojamas gana skirtingas algoritmas bei roboto kojų valdymas laiko ir padėties atžvilgiu.



12 pav. Kliūties aptikimas ir pasiruošimas ją įveikti [5]

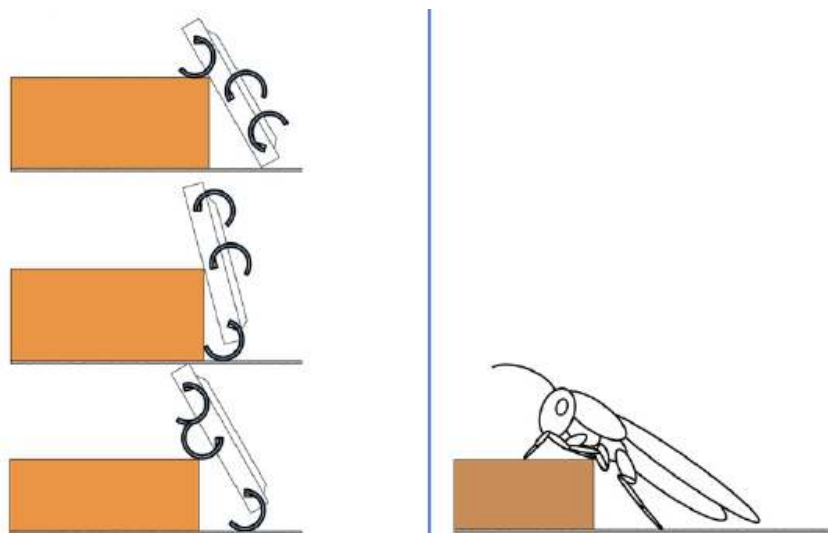
Pirmajame žingsnyje (12 pav.) robotas juda įprastiniu režimu ir horizontaliu paviršiumi. Kai aptinkama kliūtis, robotas sustoja ir pradeda pasiruošimą jos įveikimui.

Antrajame žingsnyje (12 pav.) judanti platforma turi atlikti du darbus: iškelti savo priekį kaip galima aukščiau ir arčiau kliūties, kad ją būtų galima užkabinti priekinėmis kojomis, taip pat sukongūruoti ratų ir pačio roboto pozicijas taip, kad būtų galima įveikti kliūtį. Pasiruošus lipimui, prasideda trečiasis žingsnis.



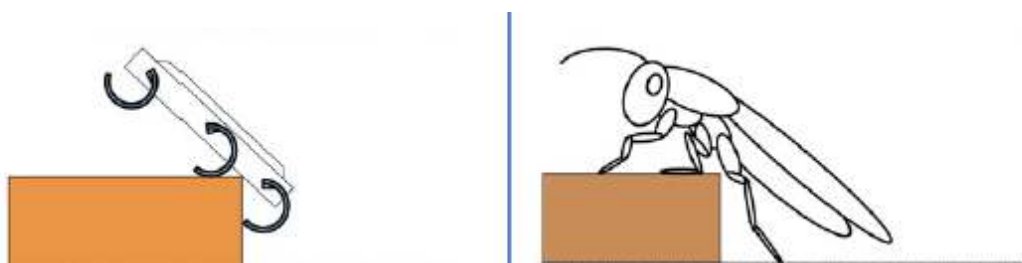
13 pav. Trečiasis žingsnis [5]

Trečiajame žingsnyje (13 pav.) robotas išmatuoja kliūtis aukštį ir įvertina ar galės efektyviai įveikti kliūtį su jau esama kojų ir kūno konfiguracija. Priklausomai nuo išmatuoto kliūtis aukščio, valdant roboto masės centrą kojomis, platformos priekinės kojos užkeliamos ant kliūtis, masės centrą pakeliant šiek tiek aukštin.



14 pav. Ketvirtasis žingsnis [5]

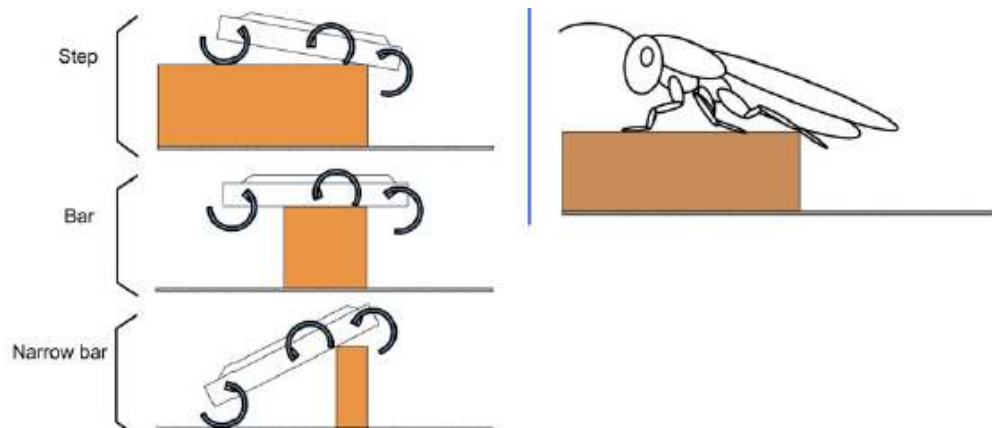
Ketvirtasis žingsnis (14 pav.) yra skirtas masės centro perkėlimui kuo galima arčiau kliūtis. Po to, kai roboto priekis buvo atremtas į kliūtis viršų trečiajame žingsnyje, vidurinėmis kojomis siekiama atsiremti į kliūtis kraštą, o galinėmis kojomis – pritraukti platformos galą kaip galima arčiau kliūtis. Šiame žingsnyje labai svarbu šiuos veiksmus atlikti itin tiksliai, kadangi pakėlus roboto priekį per aukštai, jis gali netekti pusiausvyros ir apvirsti aukštin kojomis.



15 pav. Penktasis žingsnis [5]

Penktasis žingsnis (15 pav.) – platformos užkėlimas ant kliūtis. Šiame žingsnyje vidurinės kojos kelia roboto priekinę dalį į priekį, galinės kojos stumia roboto korpusą aukštin, taip

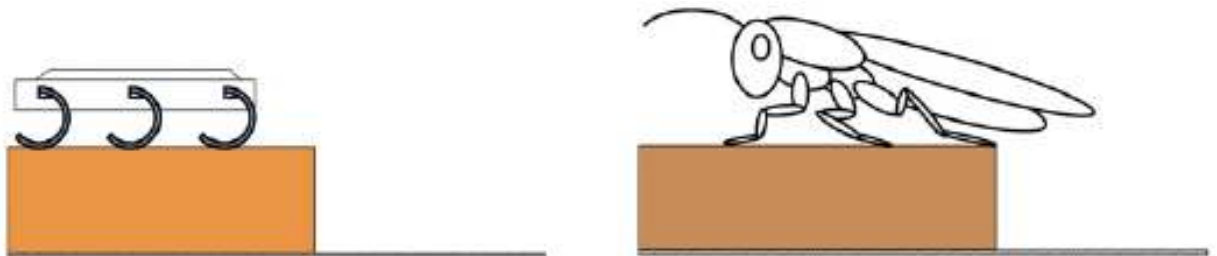
prarandant kontaktą su lygiu paviršiumi. Kol priekinės kojos neturi kontakto su kliūtis paviršiumi, jas reikia sukongigūruoti taip, kad jos būtų pasiruošusios platformos priekinės dalie kritimui ant kliūtis paviršiaus ir sugertų visą smūgį. Priekinė roboto dalis krenta ant kliūtis paviršiaus tada, kai vidurinės ir galinės platformos kojos ją pakelia taip, kad priekinės dalies sunkio jėga viršija sukuriamą vidurinių ir galių kojų jėgą ir nebegali neturėti kontakto su kliūtis paviršiumi.



16 pav. Šeštasis žingsnis [5]

Šeštajame žingsnyje (16 pav.) atliekami veiksmai priklauso nuo nustatyto kliūtis tipo. Kliūtis tipas nustatomas išmatuojant platformos posvyrio kampą ir nustatoma viena iš šių kliūtis rūšių: laiptelis (angl. *step*), kaladė (angl. *bar*) ar siaura kaladė (angl. *narrow bar*). Priklausomai nuo to, kokia kliūtis aptikta, keičiamas kojų valdymas ir perėjimas į septintą žingsnį (17 pav.):

- jei aptiktas laiptelis, tuomet ratai pasukami taip, kad platforma galėtų atsistoti ir pilna eiga judėti horizontaliu paviršiumi tolyn;
- jeigu aptikta kaladė, tuomet vidurinės kojos sukamuoju judesiu pastūmia platformą į priekį, o kai priekinės kojos pasiekia žemės paviršius, priekinės ir galinės pastūmia robotą taip, kad šis atsistotų ant žemės ir galėtų pilna eiga judėti horizontaliu paviršiumi tolyn;
- jeigu aptikta siaura kaladėlė, priekinės ir galinės pastumia robotą taip, kad šis atsistotų ant žemės ir galėtų pilna eiga judėti horizontaliu paviršiumi tolyn.

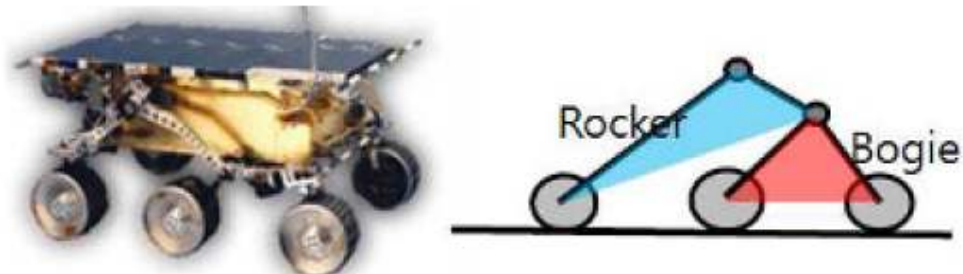


17 pav. Septintasis žingsnis [5]

1.3. „Rocker-Bogie“ tipo platformos

1.3.1. Platformos mechanika ir kinematika

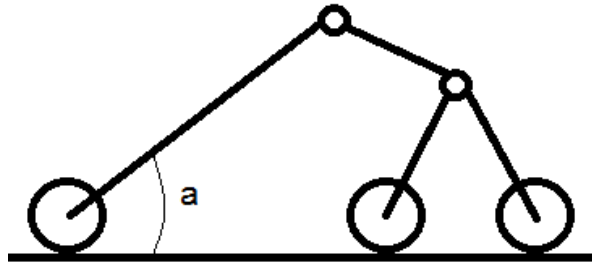
„Rocker-Bogie“ – tai sistema, sudaryta iš šešių ratų, sujungtų tarpusavyje lenktomis jungtimis. Šio tipo sistemos yra naudojamos tyrinėjant kitas planetas, kadangi sugeba gerai įveikti kelyje pasitaikiusias kliūtis ir judėti nelygiu paviršiumi [8]. „Rocker-Bogie“ tipo robotų pagrindinis rėmas yra sudarytas iš dviejų jungčių ir yra vadinamas „rocker“. Prie vieno šios konstrukcijos galo yra pritvirtinti du ratai, o prie kito – mažesnis rėmas, jungiantis keturis ratus ir vadinamas „bogie“. „Rocker“ ir „bogie“ tarpusavyje yra sujungti dviem šarnyrais, todėl gali nepriklausomai judėti vienas kito atžvilgiu. Toks mechanizmas leidžia robotui su šia važiuokle įveikti kliūtis labai gerai, tačiau tam būtina visų ratų varančioji sistema. [9]



18 pav. „Rocker-Bogie“ platforma [9]

Kadangi kiekvienas ratas yra varomas nepriklausomai, todėl turint „Rocker-Bogie“ tipo važiuoklę, kiekvienas ratas visada kontaktuoja su paviršiumi, neprarasdamas stabilumo [8]. Įveikti kliūtį kelyje nėra sudėtinga, tačiau, iškyla problemos, robotui lipant laiptais – ne visada sukuriama reikiama jėga, kai priekiniai ratai atsiremia į laiptelį. Šią problemą galima iš dalies pašalinti didinant kampą tarp „bogie“ modulyje esančių jungčių, taip padidinant lipimo galimybes ir galimą didžiausią įveikiamą laiptelio aukštį. [9]

Vienas iš svarbiausių parametru, kuriuos reikia įvertinti kuriant „Rocker-Bogie“ tipo robotą, skirtą judėti laiptais, tai yra kampas a tarp paviršiaus ir vienos iš jungčių. Kadangi lipant laiptas šis kampas keičiasi, todėl norint, kad platforma nepavirstų, jis turi būti ne didesnis kaip 90 laipsnių. Jeigu šis laipsnis yra didesnis, tuomet masės centro judėjimas lipimo metu gali viršyti leistiną ribą, o platforma apvirsti. 90 laipsnių kampas yra ribinis, todėl visada reikia įvertinti ir atsargą nenumatytiems svyravimams [9].



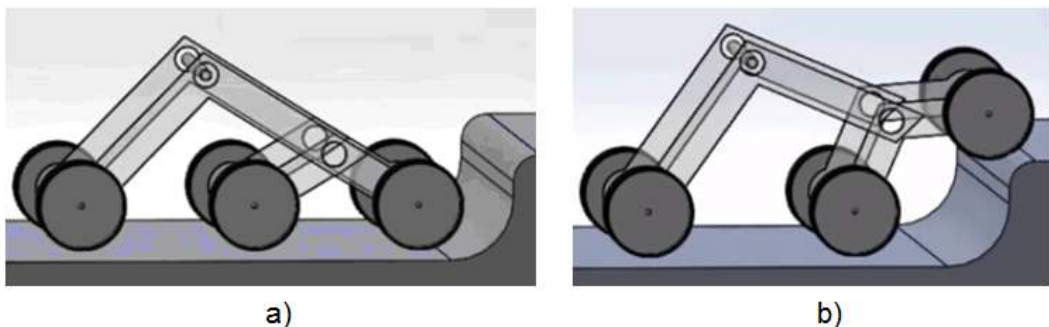
19 pav. Kampas tarp jungties ir paviršiaus

Ne visada vienus laiptus įveikianti platforma gali įveikti visų dydžiu ir tipų laiptus. Jeigu laipteliai yra per aukšti ir per trumpi, kad išlaikytų dvi poras ratų vienu metu, tuomet kilti laiptais robotas negalės. Ratais remiantis į vertikalų, ne horizontalų laiptelių paviršių, robotas nesukuria tokios jėgos, kuri galėtų pakelti į viršų jo paties svorį, o papildomas krovinys viską tik apsunkintų. [9]

„Rocker-Bogie“ tipo platformos lipimą laiptais galima suskirstyti į tris etapus.

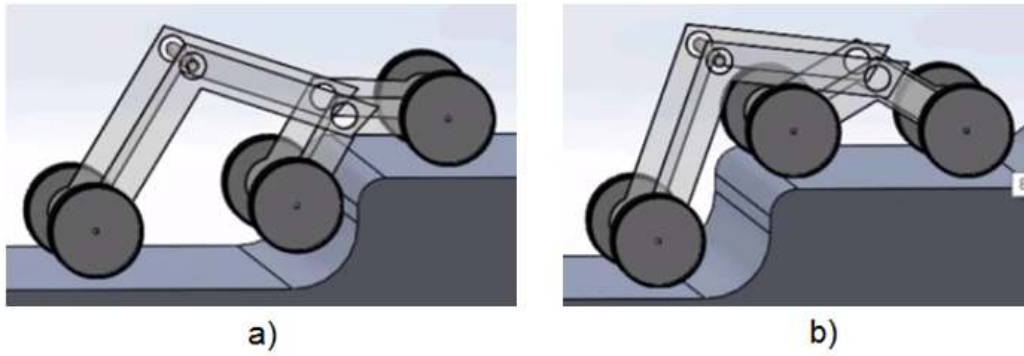
Pirmasis etapas, laiptelio vertikalaus paviršiaus įveikimas, susideda iš dviejų dalių (20 pav.). Kairėje paveikslėlio pusėje pavaizduotas pasiruošimas lipimui – platforma yra ant lygaus paviršiaus, prieš laiptelį (20 pav. a). Judant lygiu paviršiumi, gali būti varomi tik du ratai, priklausomai nuo valdymo algoritmo, tačiau geriausias valdymas gaunamas kai varomi visi ratai. Dešinėje paveikslėlio pusėje (20 pav. b) matoma, kad vadinamoji „bogie“ dalis priekiniais ratais užlipo ant laiptelio išorinės briaunos, pasikeitė kampas a (19 pav.).

Antrasis etapas prasideda tuomet, kai priekiniai „bogie“ ratai yra ant horizontalaus laiptelio paviršiaus. Tuomet šie ratai traukia platformą link sekančio laiptelio, kiti modulio dalies ratai vykdo stūmimą vertikalia kryptimi, o „rocker“ dalies ratai – horizontalia kryptimi (21 pav. a). Kai visas „bogie“ modulis atsiduria ant laiptelio horizontalaus paviršiaus (21 pav. b), tuomet jo ratai platformą traukia link sekančio laiptelio, o vis dar prieš pirmąjį laiptelį esantys „rocker“ ratai, stumia platformą horizontalia kryptimi, o jeigu ratai jau atsirėmę į laiptelį – stumia vertikalia kryptimi.



20 pav. Pirmasis etapas [10]

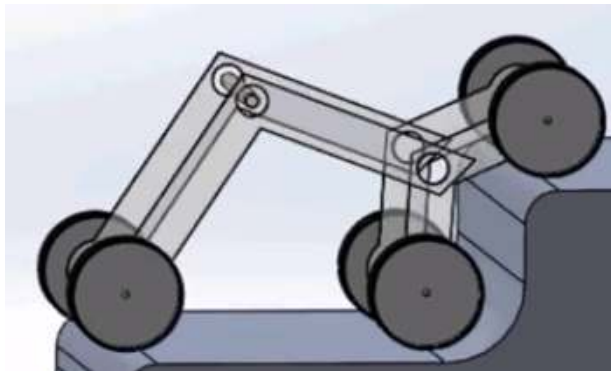
a) judėjimas horizontaliai; b) judėjimas laiptais



21 pav. Antrasis etapas [10]

a) judėjimais laiptais; b) judėjimas laiptais

Trečiajame etape (22 pav.) atliekami veiksmai gali kisti priklausomai nuo laiptų ir pačio roboto modifikacijų. Jeigu laiptelio ilgis yra ilgesnis už platformą, tuomet galiniai platformos ratai užlipa ant laiptelio ir iš naujo pradėdamas pirmasis žingsnis. Nagrinėjamu atveju, kai laiptelis yra trumpesnis už platformą, tuomet tuomet galiniais ratams lipant ant pirmojo laiptelio, priekiniai „*bogie*“ modulio ratai lipa ant antrojo laiptelio. Kai galiniai ratai visiškai užlipa ant laiptelio, pirmieji ratai jau yra užlipę ant antrojo laiptelio, todėl lipimo ciklas tęsiamas ne nuo pirmojo žingsnio, tačiau nuo antrojo žingsnio pirmosios dalies.

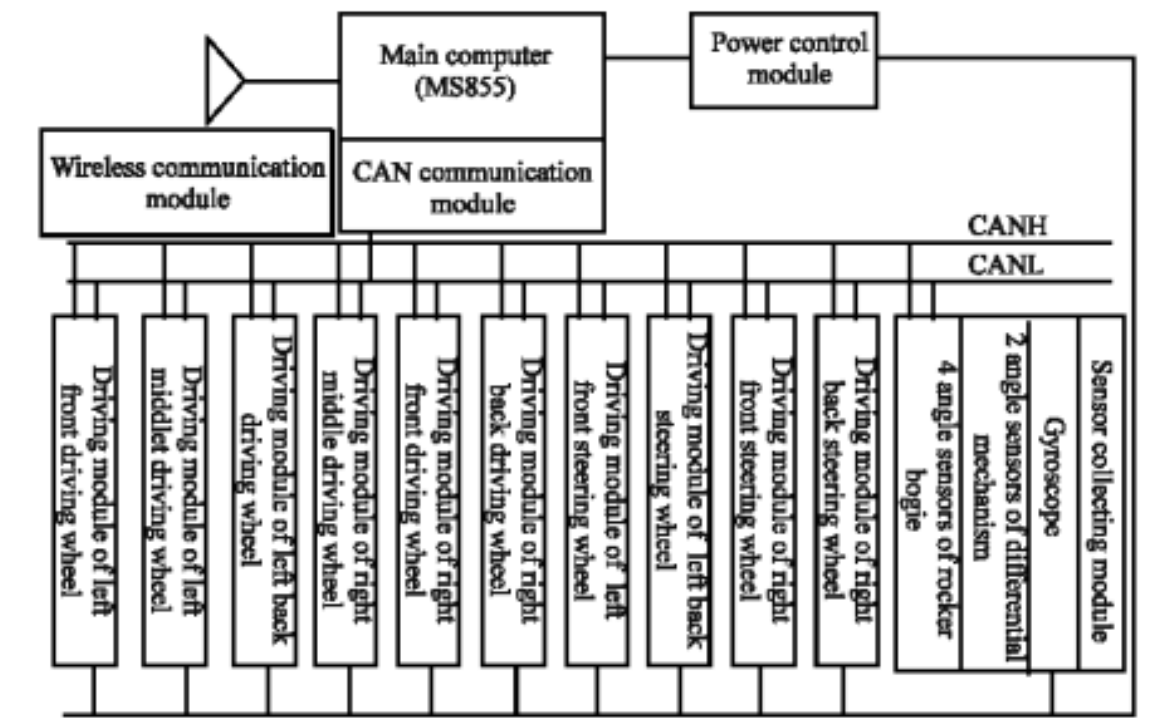


22 pav. Trečiasis etapas [10]

1.3.2. Galima valdymo sistema

Viena iš svarbiausių problemų valdant įvairius robotus, yra aparatūrinės ir programinės įrangos suderinimas. Vienas iš lengviausių būdų tai išspręsti – valdomų roboto modulių sujungimas į bendrą tinklą (23 pav.). Panaudotas CAN tinklas. [1]

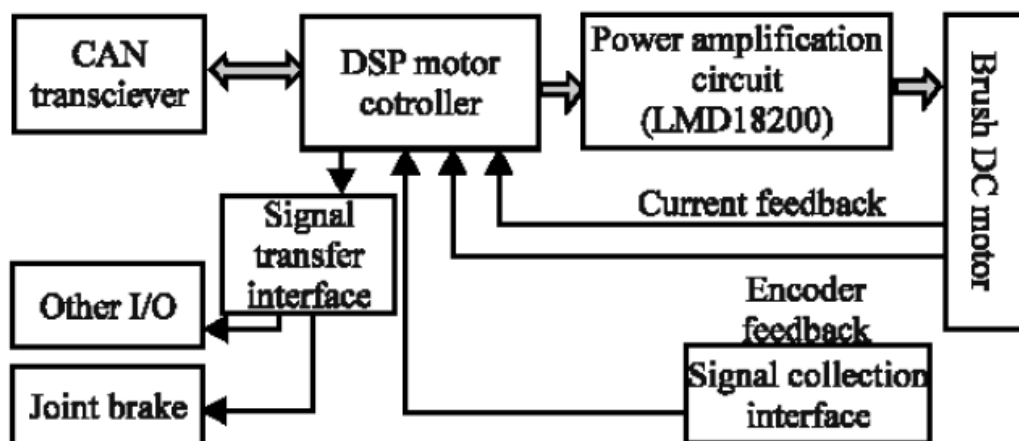
Analizuojant „rocker-bogie“ platformos vieną iš galimų valdymo modelių, reikia paminėti, kad šiame modelyje roboto ratai ne tik valdomi pozicijomis „į priekį“ ir „atgal“, tačiau taip pat ir vairuojami į šonus – platformos judėjimui naudojama 10 nuolatinės srovės elektros variklių. Į CAN tinklą sujungti ne tik elektros varikliai, tačiau ir platformos pokrypio jutikliai, giroskopas. [1].



23 pav. Platformos valdymas panaudojant CAN tinklą [11]

Tam, kad elektros variklis būtų valdomas CAN tinklo signalais, variklis turi būti valdomas skaitmeninių signalų valdikliu (angl. *DSP*) ir papildomais įtaisais. Funkcinė schema pateikta žemiau esančiame paveiksle (24 pav.). Varančiųjų ir vairuojančiųjų variklių valdymas yra skirtingas: vairuojamasis variklis valdomas gaunant informaciją iš pokrypio jutiklio apie kampą, kurio ratas yra pasuktas, o varančiojo rato valdymas turi keletą režimų. Varantysis elektros variklis gali būti valdomas gaunant informaciją apie:

- apsisukimų skaičių, panaudojant enkoderius;
- matuojant variklio inkaro srovę, kuri gali informuoti apie patiriamą apkrovą. [11]



25 pav. Nuolatinės srovės elektros variklio valdiklis [11]

1.4. Skyriaus išvados

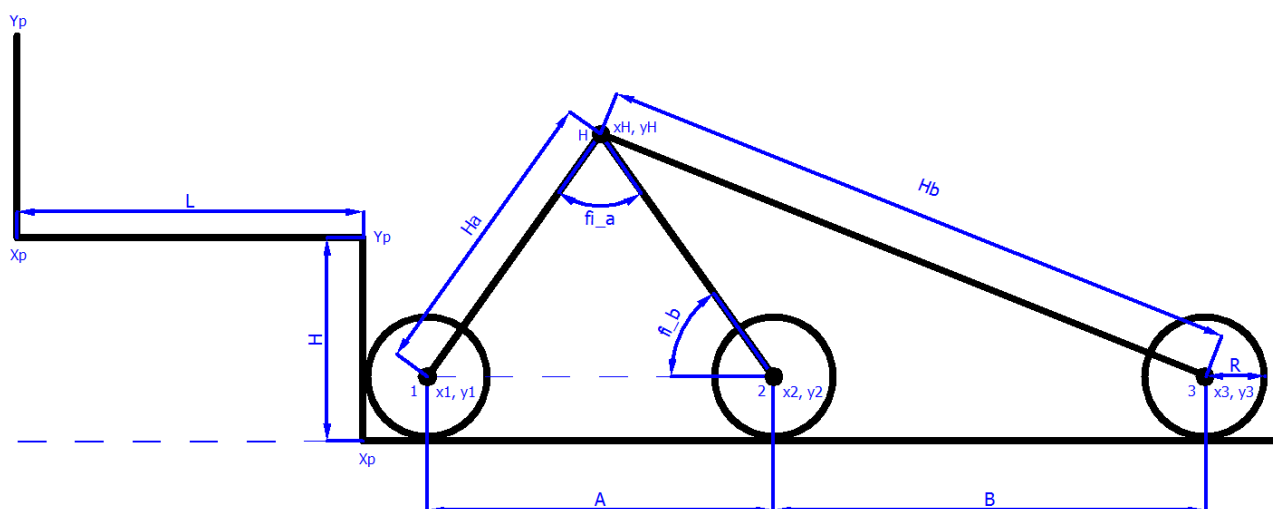
1. Sparčiai robotizuojant įvairius procesus, nuolat kuriamos ir tobulinamos įvairios platformos, perkeliančios įvairius krovinius per kliūtis ar juos užkeliančius laiptais aukštyn.
2. Daugiaračius modulius turinčios platformos įvairiuose gaminiuose naudojamos senokai, tačiau vis dar yra tobulinamos, siekiant jas pritaikyti neįgaliųjų judėjimui.
3. Robotai, kuriuose realizuotas vabzdžių judėjimas, puikiai įveikia kliūtis ir nelygius paviršius, tačiau yra gana nestabilios, todėl praktinio krovinių pernešimo pritaikymo kol kas neturi.
4. JAV kosmonautikos inžinierių sukurta „RHex“ platforma puikiai įveikia nelygius paviršius, klipa laiptais, stabiliai išlaiko prie važiuoklės pritvirtintą krovinį. Ši platforma gali būti panaudota ją pritaikant neįgaliųjų judėjimui.

2. MATEMATINĖ MOBILIOS PLATFORMOS ANALIZĖ

Šiame skyriuje matematiškai nagrinėjamas „Rocker-Bogie“ tipo mobilios platformos judėjimas spiraliniais laiptais. Matematinė analizė reikalinga siekiant sukurti ir ištirti roboto valdymo sistemą.

2.1. Sistemos aprašymas

Tiriamą sistemą sudaro tipiniai laiptai ir tipinė „Rocker-Bogie“ platformos važiuoklė. Tam, kad būtų galima aprašyti tokią sistemą, pirmiausia reikia aprašyti ją dvimatėje plokštumoje (26 pav.). Dvimatėje plokštumoje naudojamos X ir Y koordinatės.



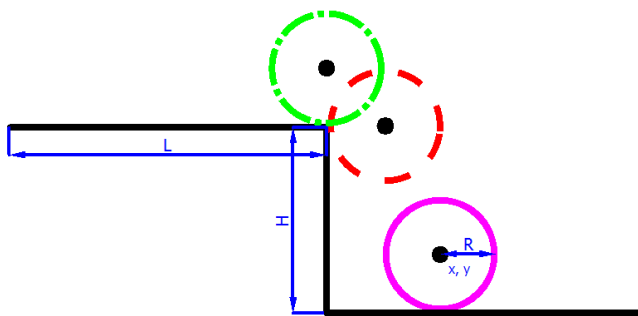
26 pav. Tiriama sistema 2D plokštumoje

Sistemos matematinio modelio konfigūracija:

- Trys identiški ratukai 1-3, kurių spindulys yra R ;
- Ratukai kampu φ_a yra standžiai sujungti dvejomis jungtimis, kurių ilgiai yra H_a ;
- Atstumas tarp ratukų 1 ir 2 yra A , o tarp 2 ir 3 – B ;
- Kampas, esantis tarp ratukų 1 ir 2 centrus jungiančios tiesės bei ratukus 1 ir 2 jungiančios standžiosios jungties kraštinės, yra φ_b ir jis yra pastovus;
- Pirmuosius du ratukus su trečiuoju per šarnyrą H jungia jungtis, kurios ilgis yra H_a ;
- Laiptelio ilgis yra H , aukštis – L ;
- Laiptelio X koordinatę aprašo dydis x_p , o Y koordinatę – y_p ;
- Ratukų 1-3 padėties aprašomos dydžiais x_n, y_n , o šarnyro H – x_H, y_H ;
- Pradinės ratukų Y koordinatės y_1, y_2, y_3 yra lygios ratuko spinduliui R .

Tam, kad „Rocker-Bogie“ ir laiptų sistemą būtų galima aprašyti matematinėmis lygtimis, būtina platformos lipimą suskaidyti į tris dalis: judėjimą horizontaliai, judėjimą vertikalčiai ir ratuko judėjimo kitimą iš horizontalaus į vertikalų (27 pav.). Horizontaliai judančio ratuko kontūras –

vientisas, judančio vertikaliai – punktyrinis, o judėjimo būsenos perėjimą iš vertikalios į horizontalų vaizduoja tarpusavyje susikirtę ratukai.

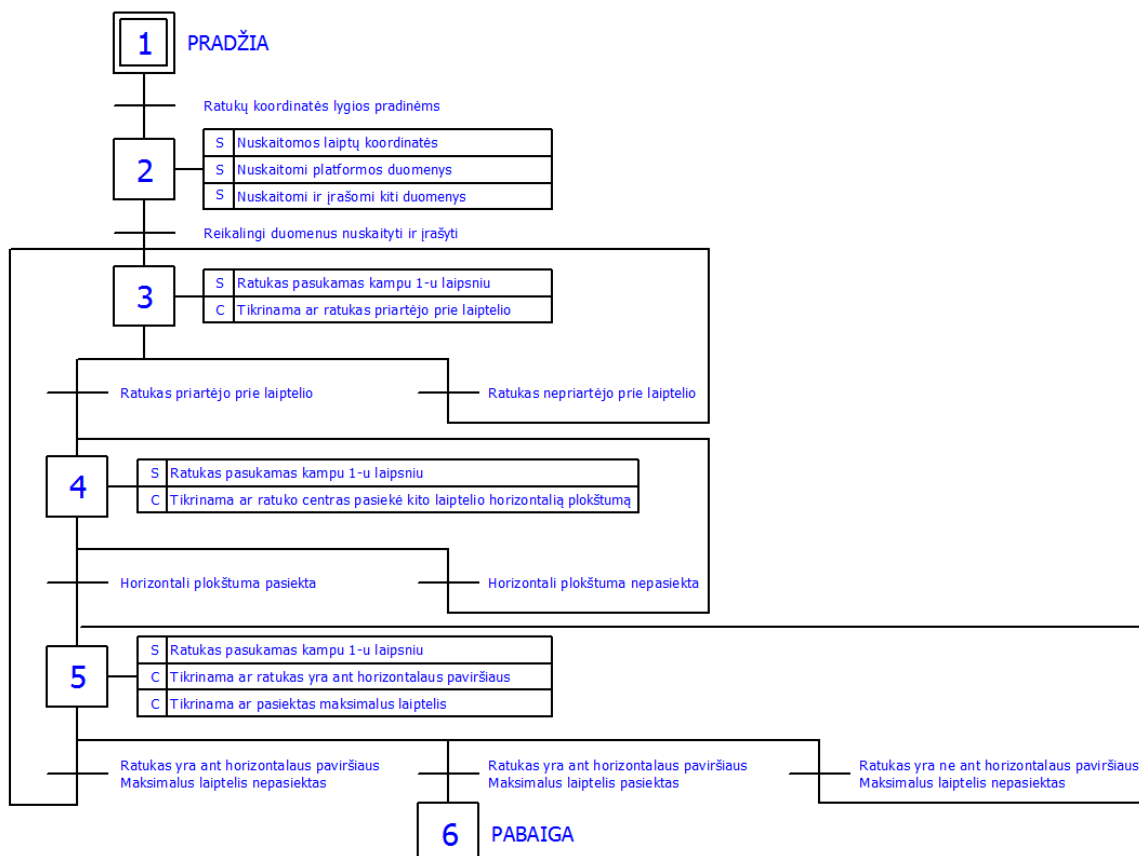


27 pav. Ratuko judėjimas laiptu

Aprašant ratukų judėjimą laiptais, patogiausia pasirinkti vieną (vedantįjį) tašką, kurio centro atžvilgiu bus nagrinėjamos kitų dviejų ratukų centrų padėties. Kadangi vedančiojo ratuko lipimas laiptais nėra sudėtingas, todėl jį galima aprašyti panaudojant *SFC* algoritimą.

2.2. Pirmojo ratuko judėjimas laiptais

Pirmojo ratuko algoritmas pateiktas žemiau esančiame supaprastintame *SFC* algoritme (28 pav.). Algoritme nepamėtos kiekviename žingsnyje naudojamos formulės tam, kad jo neperkrauti.



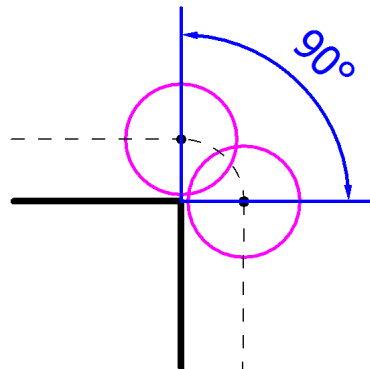
28 pav. Pirmojo ratuko lipimo laiptais supaprastintas algoritmas

Matoma, kad algoritmą sudaro keturi žingsniai ir po kiekvieno žingsnio esančios pereigos. Algoritmo žingsniai – tai veiksmai ir užduotys, skirti pradinių duomenų nuskaitymui, apskaičiavimui (antrasis žingsnis) bei pirmojo ratuko judėjimui laiptais (3 - 5 žingsniai).

Žodinis ratuko judėjimo laiptais algoritmas:

- 1) Įvykdžius pradžios sąlygą, patikrinama ar ratukai yra laiptų pradžioje ir gali jais judėti, $x_l = 0, y_l = R$.
- 2) Antrame algoritmo žingsnyje nuskaitymos laiptų koordinatės (x_l ir y_l) bei kiti įvestiniai duomenys, sudaromi reikiami duomenų masyvai; pagal ratuko spindulį surandamas atstumas ΔL , kurį ratukas nueis jį pasukus vienu laipsniu. Kai viskas atlikta, pereinama į trečią žingsnį.
- 3) Trečiasis algoritmo žingsnis pažymimas būsenos vėliavėle $Flag = 1$ – jeigu šios vėliavėlės reikšmė lygi vienetui, reiškia, kad ratukas juda horizontalia plokštuma. Šiame žingsnyje prie ratuko centro koordinatės x_l pridedama reikšmė ΔL , $x_l = x_l + \Delta L$. Ratuką pasukus vienu laipsniu patikrinama ar ratukas neatsirėmė į laiptų statmenį. Jeigu sąlyga $x_l \geq x_p - R$ netenkinama, tuomet žingsnyje esantys veiksmai ir užduotys kartojami iš naujo. Jeigu sąlyga tenkina, laikoma, kad ratukas jau atsirėmė į laiptelio stamenį ir pereinama į sekantį žingsnį.
- 4) Ketvirtasis algoritmo žingsnis žymimas būsenos vėliavėle $Flag = 2$, reiškiančia, kad ratukas juda vertikaliai. Šiame žingsnyje prie ratuko centro koordinatės y_l pridedama reikšmė ΔL , $y_l = y_l + \Delta L$. Ratuką pasukus vienu laipsniu patikrinama ar ratuko centro y_l koordinatė nėra lygi laiptelio koordinatei y_p . Jeigu sąlyga netenkinama, tuomet žingsnis kartojamas. Jeigu sąlyga tenkinama, tuomet pereinama į penktąjį algoritmo žingsnį.
- 5) Penktasis algoritmo žingsnis pažymimas būsenos vėliavėle $Flag = 3$, reiškiančia, kad ratukas iš vertikalios judėjimo pereina į horizontalų. Žemiau esančiame paveiksle (29 pav.) pavaizduotas ratuko judėjimo krypties pasikeitimas. Punktyrine linija pažymėta ratuko centro judėjimo trajektorija. Jeigu prieš tai aprašytuose judėjimo etapuose keitėsi tik viena ratuko centro koordinatė (x_l arba y_l), tai šiame judėjimo etape keičiamos abi ratuko centro koordinatės, o pats ratukas turi pasisukti 90 laipsnių kampą. Šiame žingsnyje prie ratuko centro koordinatė x_l ir y_l pridedama reikšmė ΔL , $x_l = x_l + \Delta L$, $y_l = y_l + \Delta L$. Prie centro koordinatė x_l ir y_l pridėjus po vieną reikšmę ΔL patikrinama, ar x_l ir y_l tenkina šias sąlygas: $x_l = x_p, y_l = y_p + R$. Jeigu šios sąlygos yra tenkinamos, tuomet patikrinama ar nepasiektas maksimalus laiptelių skaičius. Kadangi šiame algoritmo žingsnyje tikrinamos trys sąlygos, todėl galimos trys judėjimo laiptais baigtys:
 - a. ratukas yra ant horizontalaus paviršiaus, maksimalus laiptelis nepasiektas; pereinama į trečiąjį algoritmo žingsnį;

- b. ratukas yra ant horizontalaus paviršiaus, maksimalus laiptelis pasiektas; judėjimo laiptais pabaiga;
- c. ratukas yra ne ant horizontalaus paviršiaus; maksimalus laiptelis nepasiektas; žingsnis kartojamas iš naujo.

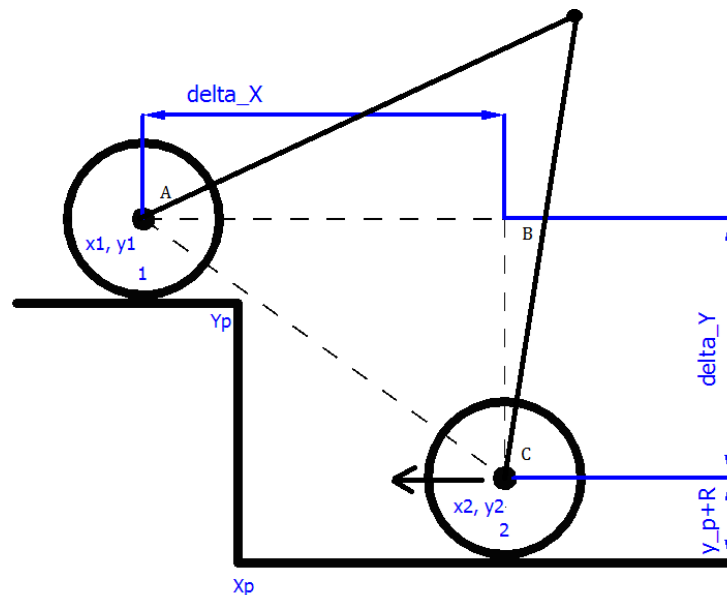


29 pav. Ratuko judėjimo krypties kitimas

2.3. Antrojo ratuko judėjimas laiptais

2.3.1. Antrojo ratuko judėjimas horizontalia plokštuma

Antrojo ir ratukų centro padėtis nustatoma pagal pirmojo ratuko centro padėtį. Tam, kad antrojo ratuko centro padėtis jam judant horizontalia plokštuma būtų apskaičiuota, nubrėžiamas supaprastintas dviejų ratukų ir laiptų modelis 2D plokštumoje (30 pav.).



30 pav. Antrojo ratuko judėjimas horizontalia kryptimi

Iš brėžinio matoma, kad antrojo ratuko centro koordinatėms x_2 ir y_2 nustatymui naudojami šie duomenys:

- Pirmojo ratuko centro koordinatės x_1 ir y_1 ;
- Laiptelio x koordinatė x_p (nulinio laiptelio koordinatė nelygi nuliui);

- Laiptelio y koordinatė y_p (nulinio laiptelio koordinatė lygi nuliui);
- Atstumas tarp antrojo ratuko centro ir horizontalios plokštumos $y_p + R$;
- Įsivaizduojamas trikampis ABC , kurio kraštinės:
 - o AB atitinka atstumą Δx tarp ratukų x koordinatė;
 - o BC atitinka atstumą Δy tarp ratukų y koordinatė;
 - o AC atitinka atstumą A tarp pirmo ir antrojo ratuko.

Antrojo ratuko koordinatė y_2 ratukui judant horizontalia plokštuma, visada lygi:

$$y_2 = y_p + R \quad (1)$$

Kadangi pirmojo ratuko koordinatė y_1 yra žinoma, todėl atstumas Δy lygus:

$$\Delta y = y_1 - (y_p + R) \quad (2)$$

Atstumas Δx surandamas panaudojant Pitagoro teoremą:

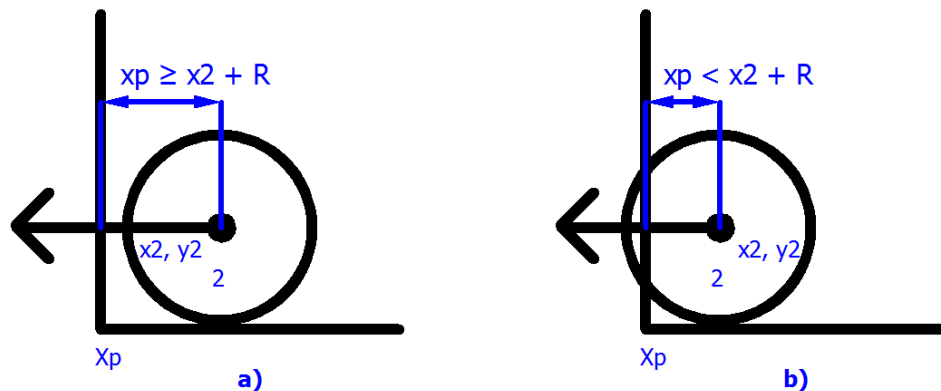
$$\Delta x = \sqrt{A^2 - (\Delta y)^2} = \sqrt{A^2 - (y_1 - (y_p + R))^2} \quad (3)$$

Žinant atstumą Δx , galima surasti antrojo ratuko koordinatę x_2 :

$$x_2 = x_1 - \Delta x \quad (4)$$

Apskaičiavus antrojo ratuko koordinates x_2 ir y_2 , tikrinama ar ratukas juda horizontaliai:

- Jeigu tenkinama sąlyga $x_p \geq x_2 + R$ (31 pav., a), tuomet laikoma, kad ratukas juda horizontaliai;
- Jeigu tenkinama sąlyga $x_p < x_2 + R$ (31 pav., b), tuomet laikoma, kad ratukas prisilietė prie laiptelio statmens, ratukas nebejuda horizontalia plokštuma, todėl turi būti tikrinamas ratuko judėjimas vertikalia kryptimi.

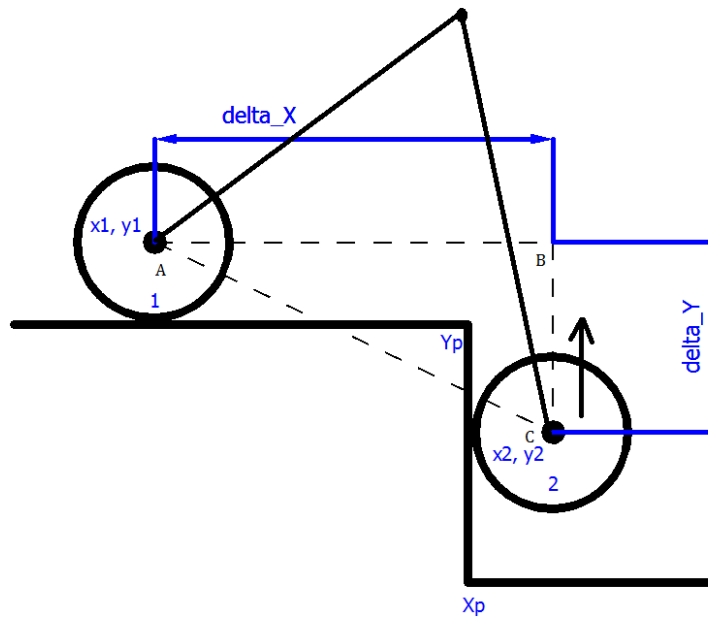


31 pav. Ratuko atsirėmimas į laiptelio vertikalią dalį

a) ratukas neatsirėmęs į laiptelį; b) ratukas atsirėmęs į laiptelį

2.3.2. Antrojo ratuko judėjimas vertikalia plokštuma

Tam, kad antrojo ratuko centro padėtis jam judant vertikaliai būtų apskaičiuota, nubrėžiamas supaprastintas dviejų ratukų ir laiptų modelis 2D plokštumoje (32 pav.).



32 pav. Antrojo ratuko judėjimas vertikalia kryptimi

Iš brėžinio matoma, kad antrojo ratuko centro koordinatė x_2 ir y_2 nustatymui naudojami šie duomenys:

- Pirmojo ratuko centro koordinatės x_1 ir y_1 ;
- Laiptelio x koordinatė x_p (nulinio laiptelio koordinatė nelygi nuliui);
- Laiptelio y koordinatė y_p (nulinio laiptelio koordinatė lygi nuliui);
- Įsivaizduojamas trikampis ABC , kurio kraštinės:
 - AB atitinka atstumą Δx tarp ratukų x koordinatė;
 - BC atitinka atstumą Δy tarp ratukų y koordinatė;
 - AC atitinka atstumą A tarp pirmo ir antrojo ratuko.

Antrojo ratuko koordinatė x_2 ratukui judant vertikaliai, visada lygi:

$$x_2 = x_p - R \quad (5)$$

Kadangi pirmojo ratuko koordinatė x_1 yra žinoma, todėl atstumas Δx lygus:

$$\Delta x = x_1 - x_2 \quad (6)$$

Atstumas Δy surandamas panaudojant Pitagoro teoremą:

$$\Delta y = \sqrt{A^2 - (\Delta x)^2} = \sqrt{A^2 - (x_1 - x_2)^2} \quad (7)$$

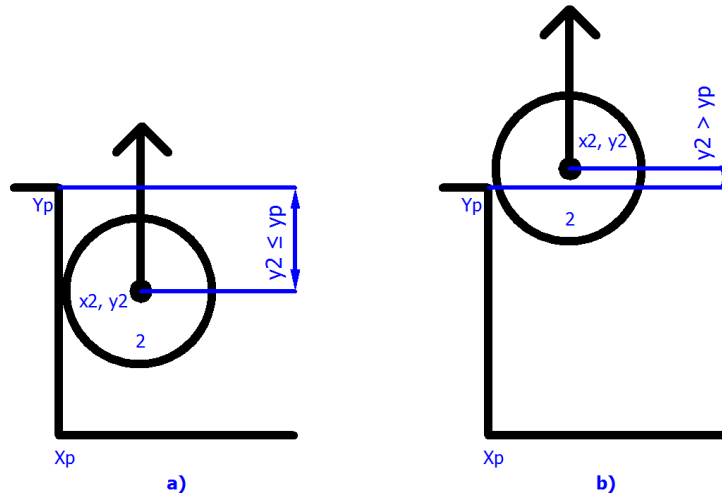
Žinant atstumą Δy , galima surasti antrojo ratuko koordinatę y_2 :

$$y_2 = y_1 - \Delta y \quad (8)$$

Apskaičiavus antrojo ratuko koordinates x_2 ir y_2 , tikrinama ar ratukas juda vertikaliai:

- Jeigu tenkinama sąlyga $y_2 \leq y_p$ (33 pav., a), tuomet laikoma, kad ratukas juda vertikaliai;

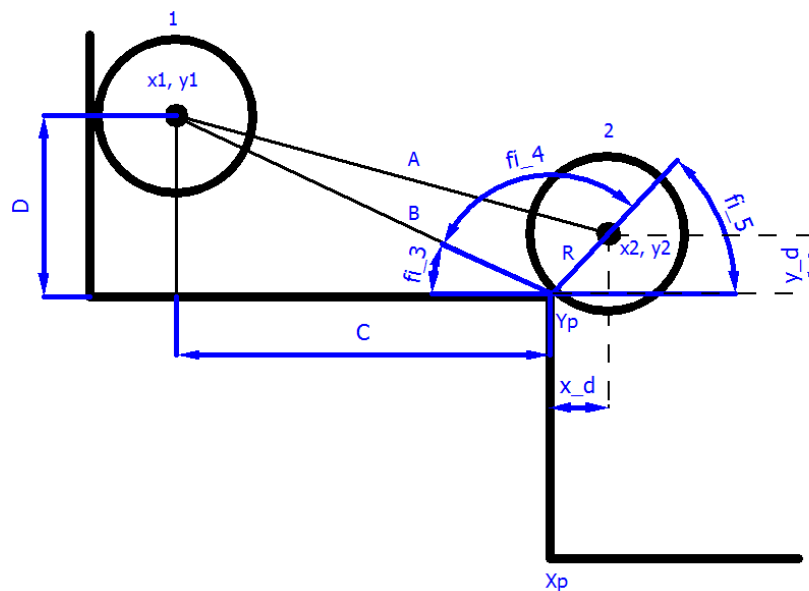
- Jeigu tenkinama sąlyga $y_2 > y_p$ (33 pav., b), tuomet laikoma, kad ratuko priartėjo prie laiptelio išorinės briaunos, ratukas nebejuda vertikaliai, todėl turi būti tikrinamas ratuko vertimasis per laiptelio išorinę briauną.



33 pav. Ratuko centro atstumas iki laiptelio išorinės briaunos
 a) ratukas neatsirėmęs į laiptelį; b) ratukas atsirėmęs į laiptelį

2.3.3. Antrojo ratuko vertimasis per išorinę laiptelio briauną

Antrojo ratuko vertimąsi per laiptelio išorinę briauną aprašo šiek tiek daugiau geometrinių skaičiavimų, nei judėjimą horizontaliai ar vertikaliai. Antrojo ratuko centro koordinatų x_2 ir y_2 nustatymui naudojami dydžiai matomi žemiau esančiame brėžinyje (34 pav.).



34 pav. x_2 ir y_2 koordinatų apskaičiavimas ratukui verčiantis per laipto išorinę laipto briauną

Iš brėžinio matoma, kad antrojo ratuko centro koordinatų x_2 ir y_2 nustatymui naudojami šie duomenys:

- Pirmojo ratuko centro koordinatės x_1 ir y_1 ;

- Laiptelio x koordinatė x_p (nulinio laiptelio koordinatė nelygi nuliui);
- Laiptelio y koordinatė y_p (nulinio laiptelio koordinatė lygi nuliui);
- Įsivaizduojami trikampiai, kurių kraštinės yra: DCB , ABR ;
- Antrojo ratuko centro atstumai iki laiptelio briaunos x_d ir y_d ;
- Kampai φ_3 , φ_4 , φ_5 , nusakantys pirmojo ir antrojo ratuko geometrinį santykį su laiptais.

Pirmiausia, remiantis Pitagoro teorema, apskaičiuojamas trikampis, kurio kraštinės yra DCB :

$$B = \sqrt{C^2 - D^2} = \sqrt{(x_p - x_1)^2 - (y_p - y_1)^2} \quad (9)$$

Apskaičiuojami kampai φ_3 , φ_4 , φ_5 , remiantis trikampių apskaičiavimui naudojamomis formulėmis ir kosinusų teorema:

$$\varphi_3 = \arcsin(D / B) \quad (10)$$

$$\varphi_4 = \arccos\left(\frac{R^2 + B^2 - A^2}{2RB}\right) \quad (11)$$

$$\varphi_5 = 180^\circ - \varphi_3 - \varphi_4 \quad (12)$$

Apskaičiuojami antrojo ratuko centro atstumai iki laiptelio briaunos x_d ir y_d :

$$\sin(\varphi_5) = y_d / R \quad (13)$$

$$y_d = R \cdot \sin(\varphi_5) \quad (14)$$

$$\cos(\varphi_5) = x_d / R \quad (15)$$

$$x_d = R \cdot \cos(\varphi_5) \quad (16)$$

Apskaičiuojamos antrojo ratuko centro koordinatės x_2 ir y_2 :

$$x_2 = x_1 - C - x_d \quad (17)$$

$$y_2 = y_1 - D - y_d \quad (18)$$

Apskaičiavus antrojo ratuko centro koordinates, tikrinama ar ratukas persivertė per laiptelio išorinę briauną. Jeigu tenkinama sąlyga $x_2 < x_p$, o $y_2 < y_p + R$, tuomet ratukas vis dar verčiasi, priešingu atveju – ratukas apsivertęs ir pradeda tikrinti trečiojo ratuko centro padėtis.

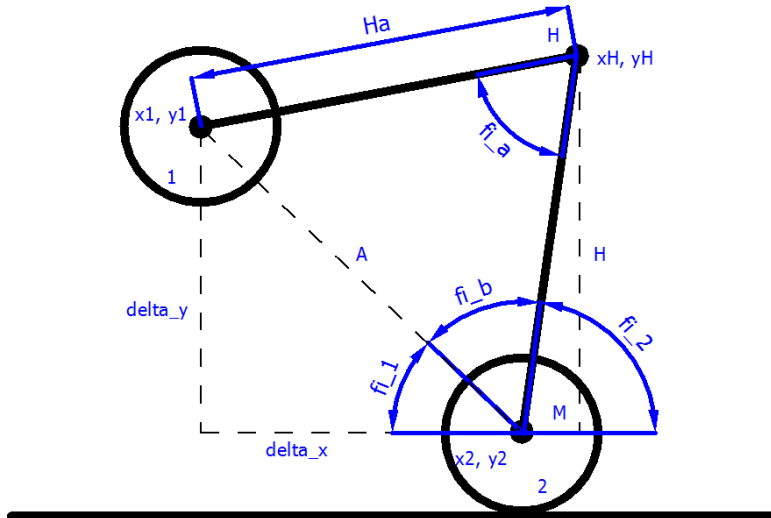
2.4. Trečiojo ratuko judėjimas laiptais

Trečiojo ratuko centro taško koordinatės x_3 ir y_3 randamos panaudojant platformos šarnyro H koordinatės x_H ir y_H (26 pav.). Naudoti šarnyrą H kaip atskaitos tašką yra patogų, kadangi atstumas tarp jo ir trečiojo ratuko niekada nekinta.

2.4.1. Šarnyro H koordinačių radimas

Iš brėžinio (35 pav.) matoma, kad šarnyro H centro koordinačių x_H ir y_H nustatymui naudojami šie duomenys:

- Pirmojo ir antrojo ratuko centro koordinatės x_1, y_1 ir x_2, y_2 ;
- Ratukus jungiančių jungčių vidiniai kampai φ_a ir φ_b ;
- Šarnyro ir ratukų centrų jungčių ilgiai H_a ;
- Įsivaizduojami trikampiai su kraštinėmis $A, \Delta x, \Delta y$ ir H_a, H, M ;
- Kampas φ_1 , kurį sudaro ratukų junginys su koordinatinių ašimi x ;
- Kampas φ_2 , kurį sudaro antrojo ratuko jungtis su koordinatinių ašimi x .



35 pav. Šarnyro centro koordinatinių x_H ir y_H nustatymas

Pirmiausia apskaičiuojamas kampas φ_1 . Šis kampas parodo kokį kampą pirmojo ir antrojo ratukų junginys sudaro su horizontaliu laiptų paviršiumi:

$$\varphi_1 = \arccos(\Delta x / A) \quad (19)$$

Apskaičiuojamas kampas φ_2 , parodantis kokį kampą pirmojo ir antrojo ratukų junginys sudaro su horizontaliu laiptų paviršiumi:

$$\varphi_2 = 180^\circ - \varphi_1 - \varphi_b \quad (20)$$

Apskaičiuojamas atstumas tarp antrojo ratuko ir šarnyro H centrų pagal x ašį:

$$\cos \varphi_2 = M / H \quad (21)$$

$$M = H \cdot \cos \varphi_2 = x_H \quad (22)$$

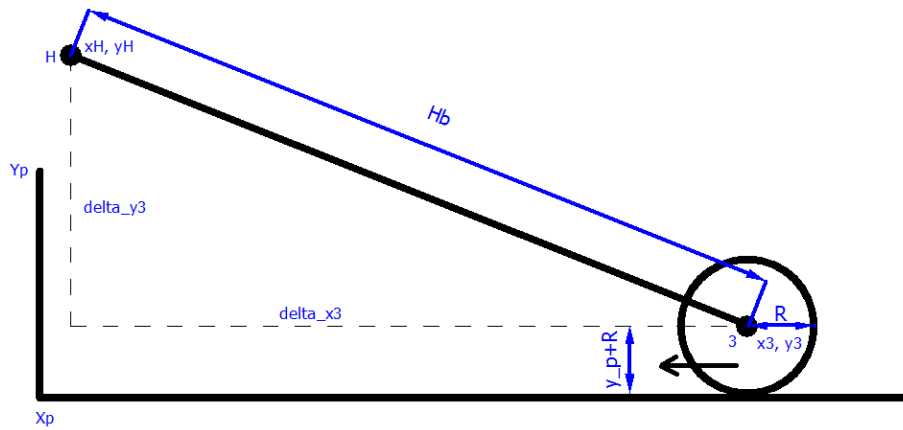
Apskaičiuojamas atstumas tarp antrojo ratuko ir šarnyro H centrų pagal y ašį:

$$\sin \varphi_2 = H / H_a \quad (23)$$

$$H = H_a \cdot \sin \varphi_2 = y_H \quad (24)$$

2.4.2. Trečiojo ratuko judėjimas horizontalia plokštuma

Trečiojo ratuko centro koordinatės x_3 ir y_3 apskaičiuojamos tokiu pat būdu, kaip ir antrojo ratuko, šiam judant horizontalia plokštuma. Nubraižomas supaprastintas brėžinys, kuriame atvaizduojami reikalingi dydžiai (36 pav.).



36 pav. Trečiojo ratuko judėjimas horizontalia kryptimi

Iš brėžinio matoma, kad naudojami ratuko centro koordinatėms x_3 ir y_3 nustatymui naudojami šie dydžiai:

- Šarnyro H centro koordinatės x_H ir y_H ;
- Šarnyrą H ir trečiojo ratuko centrą jungianti jungtis H_B ;
- Atstumas tarp trečiojo ratuko centro ir horizontalios plokštumos $y_p + R$;
- Įsivaizduojamas trikampis, su kraštinėmis H_B , Δx_3 ir Δy_3 .

Trečiojo ratuko koordinatė y_3 ratukui judant horizontalia plokštuma, visada lygi:

$$y_3 = y_p + R \quad (25)$$

Kadangi šarnyro H koordinatė y_H yra žinoma, todėl atstumas Δy_3 lygus:

$$\Delta y_3 = y_H - (y_p + R) \quad (26)$$

Atstumas Δx_3 surandamas panaudojant Pitagoro teoremą:

$$\Delta x_3 = \sqrt{H_B^2 - (\Delta y_3)^2} = \sqrt{H_B^2 - (y_H - (y_p + R))^2} \quad (27)$$

Žinant atstumą Δx_3 , galima surasti antrojo ratuko koordinatę x_2 :

$$x_3 = x_H - \Delta x_3 \quad (28)$$

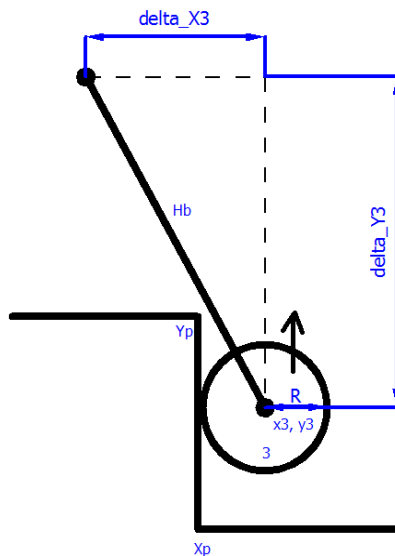
Apskaičiavus trečiojo ratuko koordinates x_3 ir y_3 , tikrinama ar ratukas juda horizontaliai:

- Jeigu tenkinama sąlyga $x_p \geq x_3 + R$ (31 pav., a), tuomet laikoma, kad ratukas juda horizontaliai (antrojo ratuko parodomasis brėžinys tinkamas ir trečiajam ratukui);
- Jeigu tenkinama sąlyga $x_p < x_3 + R$ (31 pav., b), tuomet laikoma, kad ratukas prisilietė prie laiptelio statmens, ratukas nebejuda horizontalia plokštuma, todėl turi būti tikrinamas ratuko judėjimas vertikalia kryptimi (antrojo ratuko parodomasis brėžinys tinkamas ir trečiajam ratukui).

2.4.3. Trečiojo ratuko judėjimas vertikalia plokštuma

Trečiojo ratuko centro koordinatės x_3 ir y_3 apskaičiuojamos tokiu pat būdu, kaip ir antrojo ratuko, šiam judant horizontalia plokštuma. Nubraižomas supaprastintas brėžinys, kuriame

atvaizduojami reikalingi dydžiai (37 pav.). Ratuko judėjimo laiptais galimybes gali apsunkinti ji jungiančios jungties H_b ilgis, tačiau šiame skyrelyje priimama, kad jis įtakos judėjimui neturi. Šios jungties formos ir ilgio įtaka ratuko judėjimui laiptais bus tiriama sekančiuose skyriuose.



37 pav. Trečiojo ratuko judėjimas vertikalia kryptimi

Iš brėžinio matoma, kad antrojo ratuko centro koordinatėms x_3 ir y_3 nustatymui naudojami šie duomenys:

- šarnyro H centro koordinatės x_H ir y_H ;
- šarnyrą H ir trečiojo ratuko centrą jungianti jungtis H_B ;
- laiptelio x koordinatė x_p (nulinio laiptelio koordinatė nelygi nuliui);
- laiptelio y koordinatė y_p (nulinio laiptelio koordinatė lygi nuliui);
- įsivaizduojamas trikampis, su kraštinėmis H_B , Δx_3 ir Δy_3 .

Antrojo ratuko koordinatė x_2 ratukui judant vertikaliai, visada lygi:

$$x_3 = x_p - R \quad (29)$$

Kadangi šarnyro H koordinatė y_H yra žinoma, todėl atstumas Δx_3 lygus:

$$\Delta x_3 = x_H - x_3 \quad (30)$$

Atstumas Δy_3 surandamas panaudojant Pitagoro teoremą:

$$\Delta y_3 = \sqrt{H_b^2 - (\Delta x_3)^2} = \sqrt{H_b^2 - (x_H - x_3)^2} \quad (31)$$

Žinant atstumą Δy_3 , galima surasti antrojo ratuko koordinatę y_3 :

$$y_3 = y_H - \Delta y_3 \quad (32)$$

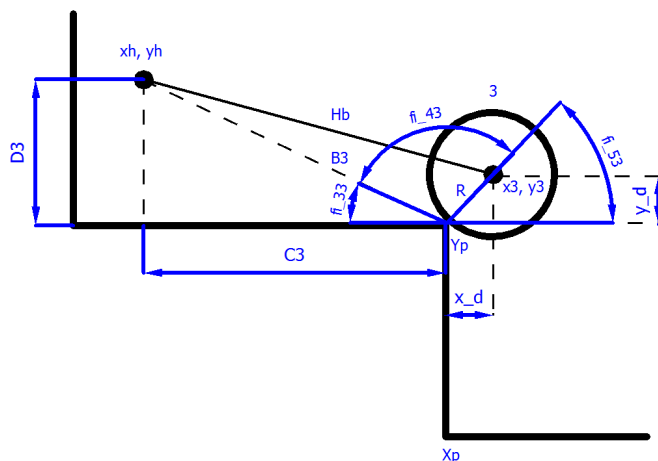
Apskaičiavus antrojo ratuko koordinates x_3 ir y_3 , tikrinama ar ratukas juda vertikaliai:

- Jeigu tenkinama sąlyga $y_3 \leq y_p$ (33 pav., a), tuomet laikoma, kad ratukas juda vertikaliai (antrojo ratuko parodomasis brėžinys tinkamas ir trečiajam ratukui);
- Jeigu tenkinama sąlyga $y_{23} > y_p$ (33 pav., b), tuomet laikoma, kad ratuko priartėjo prie laiptelio išorinės briaunos, ratukas nebejuda vertikaliai, todėl turi būti tikrinamas

ratuko vertimasis per laiptelio išorinę briauną (antrojo ratuko parodomasis brėžinys tinkamas ir trečiajam ratukui).

2.4.4. Trečiojo ratuko vertimasis per išorinę laiptelio briauną

Trečiojo ratuko vertimasi per laiptelio išorinę briauną aprašo šiek tiek daugiau geometrinių skaičiavimų, nei judėjimą horizontaliai ar vertikaliai. Trečiojo ratuko centro koordinatė x_3 ir y_3 nustatymui naudojami dydžiai matomi žemiau esančiame brėžinyje (38 pav.).



38 pav. x_3 ir y_3 koordinatė apskaičiavimas ratukui verčiantis per laipto išorinę laipto briauną

Iš brėžinio matoma, kad trečiojo ratuko centro koordinatė x_3 ir y_3 nustatymui naudojami šie duomenys:

- šarnyro H centro koordinatės x_H ir y_H ;
- šarnyrą H ir trečiojo ratuko centrą jungianti jungtis H_B ;
- laiptelio x koordinatė x_p (nulinio laiptelio koordinatė nelygi nuliui);
- laiptelio y koordinatė y_p (nulinio laiptelio koordinatė lygi nuliui);
- įsivaizduojami trikampiai, kurių kraštinės yra: $D_3C_3B_3$, H_BB_3R ;
- antrojo ratuko centro atstumai iki laiptelio briaunos x_d ir y_d ;
- kampai φ_{33} , φ_{43} , φ_{53} , nusakantys pirmojo ir antrojo ratuko geometrinį santykį su laiptais.

Pirmiausia, remiantis Pitagoro teorema, apskaičiuojamas trikampis, kurio kraštinės yra $D_3C_3B_3$:

$$B_3 = \sqrt{C_3^2 - D_3^2} = \sqrt{(x_p - x_H)^2 - (y_p - y_H)^2} \quad (33)$$

Apskaičiuojami kampai φ_{33} , φ_{43} , φ_{53} , remiantis trikampių apskaičiavimui naudojamomis formulėmis ir kosinusų teorema:

$$\varphi_{33} = \arcsin(D_3 / B_3) \quad (34)$$

$$\varphi_{43} = \arccos\left(\frac{R^2 + B_3^2 - H_B^2}{2RB_3}\right) \quad (35)$$

$$\varphi_{53} = 180^\circ - \varphi_{33} - \varphi_{43} \quad (36)$$

Apskaičiuojami antrojo ratuko centro atstumai iki laiptelio briaunos x_d ir y_d :

$$\sin(\varphi_{53}) = y_d / R \quad (37)$$

$$y_d = R \cdot \sin(\varphi_{53}) \quad (38)$$

$$\cos(\varphi_{53}) = x_d / R \quad (39)$$

$$x_d = R \cdot \cos(\varphi_{53}) \quad (40)$$

Apskaičiuojamos antrojo ratuko centro koordinatės x_3 ir y_3 :

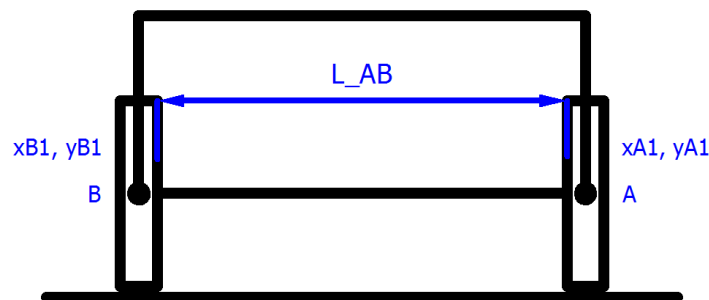
$$x_3 = x_H - C - x_d \quad (41)$$

$$y_3 = y_H - D - y_d \quad (42)$$

Apskaičiavus antrojo ratuko centro koordinates, tikrinama ar ratukas persivertė per laiptelio išorinę briauną. Jeigu tenkinama sąlyga $x_3 < x_p$, o $y_3 < y_p + R$, tuomet ratukas vis dar verčiasi, priešingu atveju – ratukas apsivertęs ir pradeda tikrinti trečiojo ratuko centro padėtis.

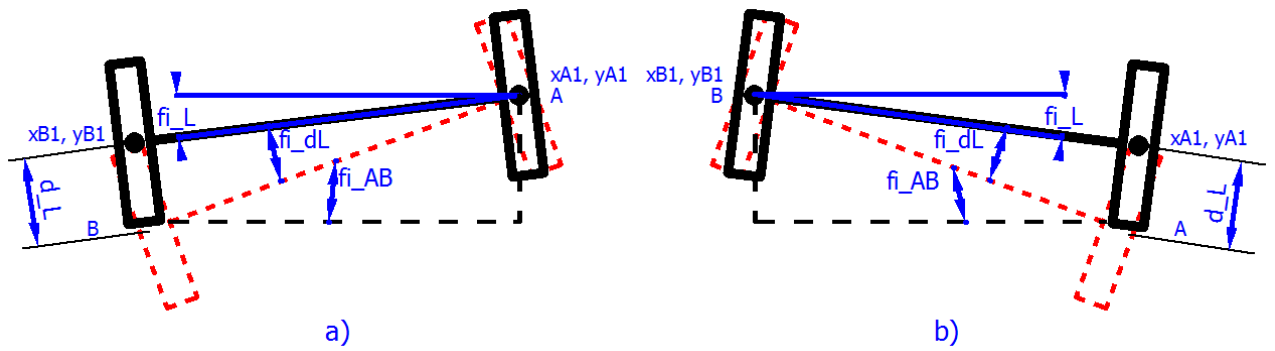
2.5. Platformos judėjimo netolygumas

Platformos judėjimą užtikrina trys varančiosios ašys, kurių vaizdas iš priekio pavaizduotas žemiau esančiame paveiksle (39 pav.). Priimama, kad kairieji ratukai žymimi papildomu indeksu A , o dešinieji – B , atstumas tarp jų centrų L_{AB} . Dėl įvairiausių veiksnių, tokių kaip ratų būklė, ratų sukimosi greičių nevienodumas, laiptų spirališkumas ir pan., platformos ratų ašis niekada nebus idealiai statmena laipteliui, ratukų A ir B atstumai iki laiptelio skirsis, todėl judėjimas laiptais bus šiek tiek įstrižas, t.y. $x_{B1} \neq x_{B2}$ ir $y_{B1} \neq y_{B2}$. Judėjimo įstrižumą reikia įvertinti tam, kad ratų ašis būtų kuo galima stamesnė laipto statmeniui.



39 pav. Platformos priekinis vaizdas

Ratų ašies judėjimo įstrižumas gali būti apibūdinamas žemiau esančiu brėžiniu (40 pav.) ir jame esančiais dydžiais. Gali būti du važiuklės įstrižumo variantai: kuomet važiuklė sukasi apie rato A centrą (40 pav. a.) ir kuomet sukasi apie rato B centrą (40 pav. b). Važiuklės padėtis apibrėžiama simetrijos ašies kampu.



40 pav. Posvyrio parametrų apskaičiavimas

a) važiuklė sukasi apie rato A centrą; b) važiuklė sukasi apie rato B centrą

Iš brėžinio matoma, kad važiuklės padėčiai nustatyti naudojami šie parametrai:

- Kampas φ_{AB} , kuriuo važiuklė yra pasvirusi;
- Kampas φ_L , kuriuo važiuklė pirmą kartą nukrypo nuo idealiai horizontalios padėties;
- Kampas φ_{dL} tarp senosios ir naujosios važiuklės padėties.

Išvardintų kampų apskaičiavimui priimama, kad skaičiuojama pirmosios ratų ašies atžvilgiu.

Apskaičiuojant kampus kitai ašiai, reikia pakeisti antrąjį ratuko indeksą, pvz. x_{B2} .

Kampas φ_{AB} apskaičiuojamas:

$$\varphi_{AB} = \arctg\left(\frac{x_{B1} - x_{A1}}{y_{B1} - y_{A1}}\right) \quad (43)$$

Kampas φ_{dL} apskaičiuojamas:

$$\varphi_{dL} = \frac{d_L}{L_{AB}} \quad (44)$$

Kampas φ_L apskaičiuojamas:

$$\varphi_L = \frac{\pi}{2} - \left[\left(\frac{\pi}{2} - \varphi_{AB} \right) + \varphi_{dL} \right] = \varphi_{AB} - \varphi_{dL} \quad (45)$$

2.6. Skyriaus išvados

1. Matematinis „Rocker-Bogie“ platformos lipimo laiptais modelis sudarytas skaičiuojant ratukų centrų koordinates vedančiojo, pirmojo, ratuko atžvilgiu.
2. Dėl platformos visų varančiųjų ratų judėjimo nevienodumo ir judėjimo netiesiais laiptais ir kitų veiksnių, atsiranda platformos važiuklės nuokrypiai nuo horizontalios padėties, kurių įvertinimas reikalingas modeliuojant judėjimą spiraliniais laiptais.
3. Matematinė platformos judėjimo spiraliniais laiptais analizė yra pagrindas, siekiant atlikti kompiuterinę platformos simuliaciją.

3. PLATFORMOS VALDYMAS PAGAL LAIPTŲ FORMĄ

„*Rocker-Bogie*“ tipo platformos judėjimą spiraliniais laiptais galima valdyti įvairiais būdais. Vienas iš jų – platformos ratukų greičių sinchronizavimas. Šiame skyriuje aprašomas platformos ratų sukimosi greičių valdymas pagal laiptų formą. Tai reiškia, kad platformos ratukai sukasi nevienodu greičiu, priklausančiu nuo atstumo iki laiptų statmens, išorinės briaunos, likusio atstumo iki jų. Nustatyti laiptų formą galima įvairiais techniniais būdais – naudoti atstumo, priartėjimo jutiklius prie kiekvieno ratuko, matuojant elektros variklio inkaro srovę ir panašiai. Priimama, kad platforma yra lanksti arba jos ratų posūkio kampai gali būti kintantys, siekiant gauti geresnį judėjimą spiraliniais laiptais.

3.1. Programos algoritmas

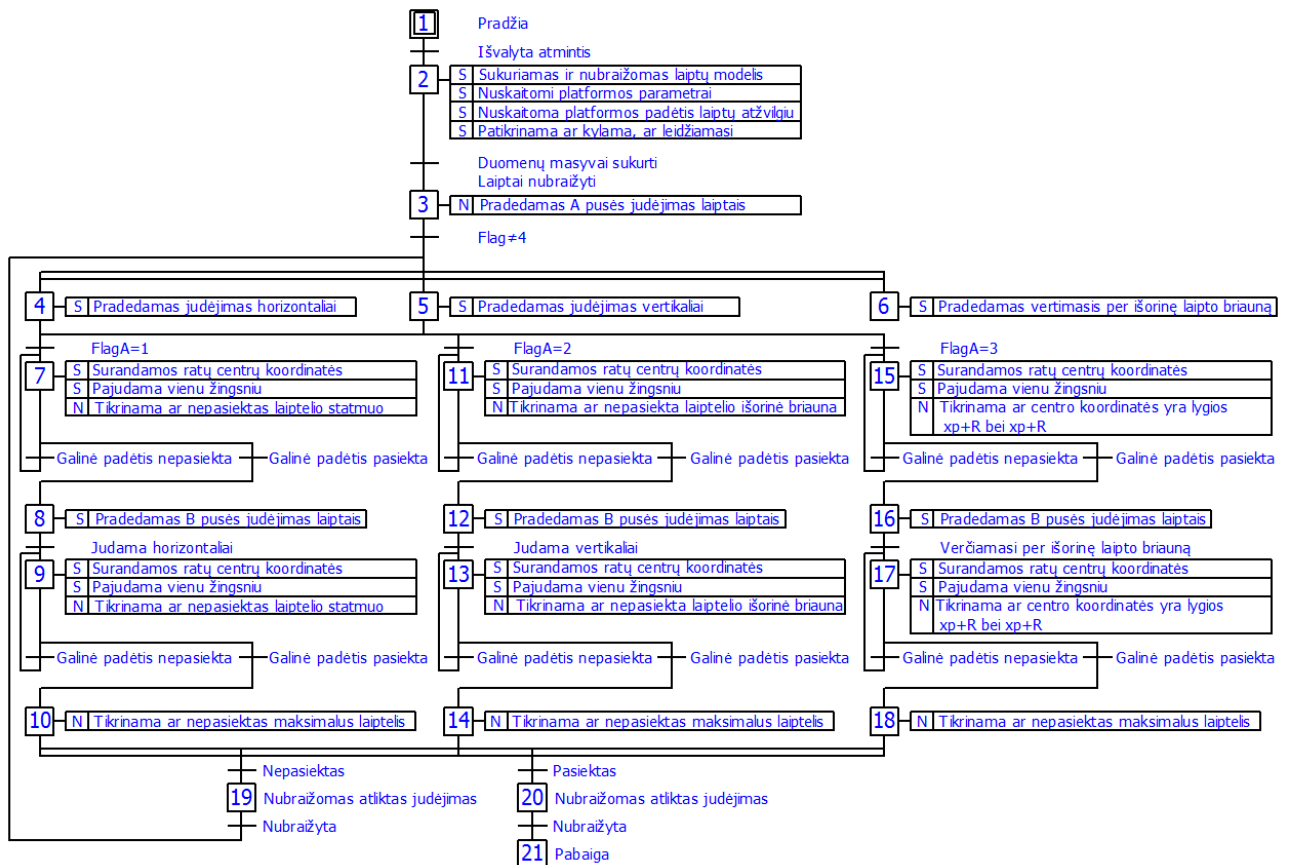
Norint „*MATLAB*“ programinio paketo aplinkoje sukurti platformos judėjimo spiraliniais laiptais simuliacinį modelį, pirmiausia reikia sukurti algoritmą, pagal kurį bus rašomas programos kodas. Programos algoritme neatsižvelgiama į technines priemones, skirtas nustatyti laiptų formai, o naudojamas apibendrintas principas – ratukas priartėjo arba ne. Algoritmo sudarymui pasirinkta *SFC* diagrama (41 pav.).

Platformos valdymui pagal laiptų formą, sukuriamos programos struktūra:

- Spiralinį laiptų aprašymas;
- Įvestinių duomenų sukūrimas;
- Pirmosios platformos pusės ratukų judėjimas horizontaliai, vertikalčiai, vertimasis per laipto išorinę briauną;
- Antrosios platformos pusės ratukų judėjimas horizontaliai, vertikalčiai, vertimasis per laipto išorinę briauną;
- Platformos judėjimo braižymas trimatėje erdvėje.

Iš algoritmo matoma, kad paleidus simuliaciją, iš pradžių išvaloma darbinė atmintis, kad programa veiktų sklandžiai be nereikalingų šiukšlių ar buvusių simuliacijų liekanų. Išvalius atmintį, atliekamas pasiruošimas platformos judėjimui. Pasiruošimo etape sukuriami ir nubraižomi spiraliniai laiptai (42 pav.).

Nubraižius spiralinius laiptus, nustatomas platformos padėties ir laiptų tarpusavio ryšys. Reikia pažymėti, kad platformos judėjimas pradedamas ne nuo laiptų pradžios, o paliekant nurodytą atstumą iki jų. Tai yra daroma siekiant judėjimo cikliškumą išlaikyti nuoseklų (judėjimas horizontaliai, vertikalčiai ir vertimasis per laipto išorinę briauną).



41 pav. *MATLAB* programos algoritmas

Programai sukūrus reikiamus duomenų masyvus, pradedamas platformos simuliacinis lipimas spiraliniais laiptais. Matoma, kad vienu metu vykdomi trys lygiagrečios programos kodo dalys: judėjimas horizontalia plokštuma, judėjimas vertikalia plokštuma ir ratuko vertimasis per išorinę laiptelio briauną.

Iš pradžių judėjimas vykdomas ratukams A . Kai ratukai pajudinami vienu žingsniu, patikrinama ar nepasiektos ribinės reikšmės, kurios kiekvienam ratukui ir judėjimo tipui yra aprašytos 5-ajame skyriuje. Jeigu šios ribos nepasiektos, reiškia, kad ratukai vis dar juda nustatyta kryptimi, priešingu atveju – reikiamas judėjimas atliktas ir pradedama ieškoti ratukų B centro koordinatų pokyčio. Jų judėjimo aprašymui tinka tokios pat formulės, kaip ir ratukams A . Vienai ašiai atlikus nustatytą judėjimą, patikrinama ar nepasiektas galinis spiralinių laiptų laiptelis. Jeigu laiptelis pasiektas – nubraižomas atliktas ratukų centrų judėjimas ir užbaigiama simuliacija, priešingu atveju – laiptų modelio lange nubraižomas įvykęs ratukų centrų padėčių pokytis, judėjimas pradedamas iš naujo, atskaitos tašku laikant apskaičiuotas ratukų centrų koordinatas.

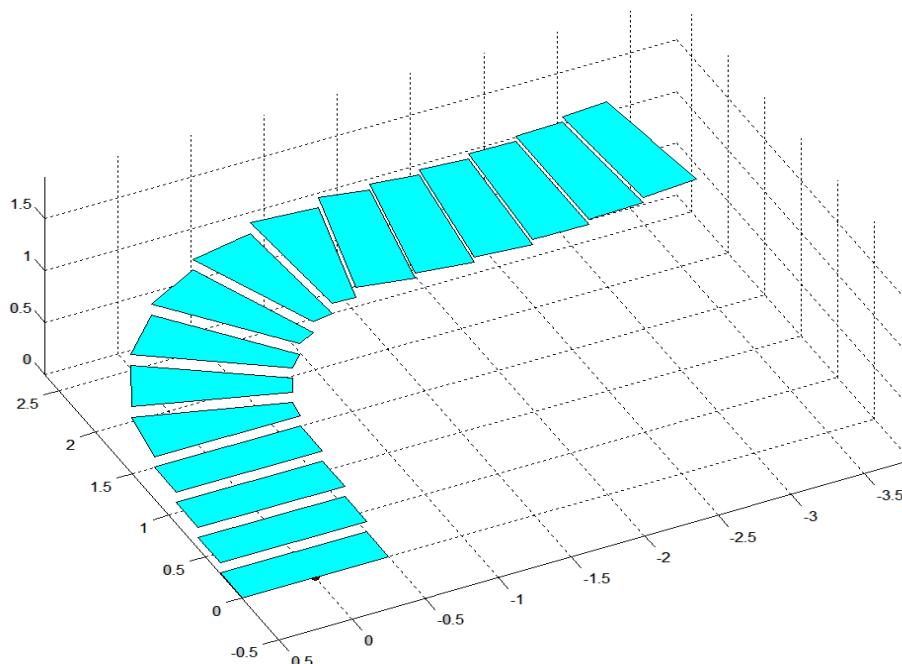
3.2. Spiralinių laiptų aprašymas

Sukuriamas trimatis laiptų modelis, kuriame būtų tyrimui reikalinga laiptų forma. Modelyje sukurtas posūkis, kuris leidžia įvertinti ar tiriama platforma gali judėti spiraliniais laiptais, ar ne.

Laiptų konfigūracija aprašoma žemiau esančioje programinio kodo dalyje:

```
spir = [ 0, 2, 0.1, 0.300;  
        -1.0, 9, 0.100, 0.300;  
        6.0, 6, 0.050, 0.350];  
NSmax = size(spir); % laiptų segmentų kiekis  
LL = 0.500; % pakopos pusplotis
```

Duomenų masyvą *spir* sudaro trys eilutės ir keturi stulpeliai. Kadangi spiralinius laiptus sudaro keletas skirtingų formų laiptų, todėl kiekviena forma yra aprašoma nauja eilute. Kiek eilučių, tiek skirtingų formų galima gauti vienuose laiptuose. Stulpeliai nurodo laiptų tiesiškumą (0 – tiesiniai), pakopų skaičių, jų aukštį ir gylį. Kintamasis *LL* saugo pakopos pusplotio reikšmę.



42 pav. Spiralinių laiptų modelis

3.3. Įvestinių duomenų nuskaitymas

Įvestinių duomenų programos dalyje aprašoma tiriamą platformą, skaičiavimo žingsnių kiekis ir kiekvienos platformos ašies santykis su spiraliniais laiptais, prieš pradėdant platformai judėti.

Žemiau esančioje programinio kodo ištraukoje aprašomas priekinės platformos ašies santykis su laiptų pradžia. Toks santykio aprašymas būtinas kiekvienai ašiai, todėl iš viso jų turi būti trys. Taip pat reikalingas iš šarnyro *H* aprašymas, kadangi jo atžvilgiu skaičiuojamos trečiosios ašies ratukų centro koordinatės.

```
fABAdeg = 0.0 ; % platformos kampas atžvilgiu pirmo laipto briaunos,  
keičiasi priklausomai nuo algoritmo
```



```

fAB1 = fAB1deg*pi/180.0 ; % kampas fAB, išreikštas radianais
LA10 = -0.5 ; LB10 = -0.5 ; % A,B mazgo atstumas nuo pradžios kiekvienam
ciklui
LA1(N) = LA0 ; L1B(N) = LB10 ; % A,B padėtys X kryptimi, judama X kryptimi
HA1(N) = R ; HB1(N) = R ; % A, B padėtis Y kryptimi
% pradinėje padėtyje B mazgas gali būti arčiau ar toliau laipto
XA1(N) = LA10; XB1(N) = XA1(N)+LAB1*sin(fAB1);
YA1(N) = -LAB1/2.0; YB1(N) = YA1(N)+LAB1*cos(fAB1);
% aprašoma pirmojo ratuko padėtis
LR1(N) = 0; % padėtis X kryptimi
HR1(N) = R; % padėtis Y kryptimi
XR1(N) = XA1(N); % padėtis X kryptimi
YR1(N) = YA1(N); % padėtis X kryptimi
deltaAB1(N) = 0.0; fiAB1(N) = 0.0; %LA(N)-LB(N);
LB100 = LB10; fiDlA1=0.0 ;

```

3.4. Judėjimas horizontaliai

Šioje programos dalyje aprašomas platformos ratuko judėjimas horizontalia plokštuma. Žemiau esantis programinio kodo fragmentas naudojamas priekinės platformos ašies, A pusės ratuko judėjimo horizontaliai aprašymui. B pusės ratuko aprašymui naudojamas tas pats kodas, tik pakeičiamas kintamųjų pavadinimuose esantis indeksas iš A į B .

```

if Flag1 == 1 % judėjimas horizontaliai
    fiA1 = fiA1 + dfA1 ; % naujas kampas sukantis aplink ratuko ašį
    LA1(N) = fiA1*R + LA10 ; % A ratuko centro padėtis
    fAB1=atan((XA1(N-1)-XB1(N-1))/(YA1(N-1)-YB1(N-1))); % platformos kampas
    fiDlA1=dfA1*R/LAB1; % posūkio kampas aplink B
    fiLL1=fAB1-fiDlA1;
    XA1(N)=XB1(N-1)-LAB1*sin(fiLL1); % ratuko A centro X
    YA1(N)=YB1(N-1)-LAB1*cos(fiLL1); % ratuko A centro Y
    LC11(N) = LA1(N);
    XC11(N)=XA1(N);
    YC11(N)=YA1(N);
    HC11(N) = R + CN(LaiptoNr_1,3); % ratuko centro aukštis
    HSN1(N)= CN(LaiptoNr_1,3);
    if XC11(N) >= CN(LaiptoNr_1+1,1) - LstopA1 % riba, kai ratukas atsiremia
        if LaiptoNr_1 < LaiptuKiekis
            LaiptoNr_1 = LaiptoNr_1 + 1 ;
        else
            Flag1=4;
        end
        LA11 = LA1(N); % taško A padėtis prieš sukantis
        XA11=XA1(N); % atsimenama padėtis prieš sukantis
        YA11=YA1(N);
        LC111 = LC11(N); % rato centro padėtis prieš sukantis ant laipto
        XC111=XC11(N);
        YC111=YC11(N);
        fiA1 = 0.0 ; %
        Flag1 = 11; % laukiama, kol kita pusė privažiuos
    end % pabaiga XA(N) >= L1M(LaiptoNr_1+1) - LstopA
elseif Flag1 == 11 % laukiama B iki atramos, stovima
    XA1(N)=XA1(N-1);
    YA1(N)=YA1(N-1);
    XC11(N)=XC11(N-1);
    YC11(N)=YC11(N-1);

```

```

LC11(N) = LC11(N-1);
LA1(N) = LA1(N-1);
end

```

Žemiau esančioje programinio kodo dalyje pateiktas fragmentas programos, kurioje aprašomas antrojo A ratuko centro koordinatėms nustatymas. Šio ratuko centro koordinatėms naudojamos matematinėje analizė gautos formulės.

```

YP1(N)=CN(NA,1)=CN(NA-1,1)+dz;
YA2(N)=YP1+R; % ratuko A centro Y
deltaY(N)=YA11-(YP1(N)+R);
deltaX(N)=sqrt(LP1^2-(deltaY(N))^2);
XA2(N)=YA1(N)-deltaX(N); % ratuko A centro X

```

3.5. Judėjimas vertikaliai

Šioje programos dalyje aprašomas platformos ratuko judėjimas vertikalia plokštuma. Žemiau esantis programinio kodo fragmentas naudojamas priekinės platformos ašies, A pusės ratuko judėjimo vertikaliai aprašymui. B pusės ratuko aprašymui naudojamas tas pats kodas, tik pakeičiamas kintamųjų pavadinimuose esantis indeksas iš A į B .

```

if Flag1 == 2 % judėjimas vertikaliai
    fiA1 = fiA1 + dfA1 ; % naujas kampas sukantis aplink ratuko ašį
    LA1(N) = fiA1*R + LA10 ; % A ratuko centro padėtis
    fAB1=atan((XA1(N-1)-XB1(N-1))/(YA1(N-1)-YB1(N-1))); % platformos kampas
    fiDLA1=dfA1*R/LAB1; % posūkio kampas aplink B
    fiLL1=fAB1-fiDLA1;
    XA1(N)=XB1(N-1)-LAB1*cos(fiLL1); % ratuko A centro X
    YA1(N)=YB1(N-1)-LAB1*sin(fiLL1); % ratuko A centro Y
    LC11(N) = LA1(N);
    XC11(N)=XA1(N);
    YC11(N)=YA1(N);
    HA11(N) = R + CN(LaiptoNr_1,3); % mazgo A aukstis
    HC11(N) = R + CN(LaiptoNr_1,3); % ratuko centro aukstis
    HSN1(N) = CN(LaiptoNr_1,3);
    if XC11(N) >= CN(LaiptoNr_1+1,1) - LstopA1 % riba, kai ratukas atsiremia
        if LaiptoNr_1 < LaiptuKiekis
            LaiptoNr_1 = LaiptoNr_1 + 1 ;
        else
            Flag1=4;
        end
        LA11 = LA1(N); % tasko A padėtis prieš sukantis
        XA11=XA1(N); % atsimenama padėtis prieš sukantis
        YA11=YA1(N);
        LC111 = LC11(N); % rato centro padėtis prieš sukantis ant laipto
        XC111=XC11(N);
        YC111=YC11(N);
        fiA1 = 0.0 ; %
        Flag1 = 11; % laukiama, kol kita pusė privažiuos
    end % pabaiga XA(N) >= L1M(LaiptoNr_1+1) - LstopA
elseif Flag1 == 11 % laukiame B iki atramos, stovime
    XA1(N)=XA1(N-1);
    YA1(N)=YA1(N-1);
    XC11(N)=XC11(N-1);

```

```

YC11(N)=YC11(N-1);
LC11(N) = LC11(N-1);
LA1(N) = LA1(N-1);
end

```

Žemiau esančioje programinio kodo dalyje pateiktas fragmentas programos, kurioje aprašomas antrojo A ratuko centro koordinatėms nustatymas. Šio ratuko centro koordinatėms naudojamos matematinėje analizė gautos formulės.

```

XP2(N)=CN(NA,1)=CN(NA-1,1)+dz;
XA2(N)=XP2(N)-R; % ratuko A centro X
deltaX(N)=XA1(N)-XA2(N);
deltaY(N)=sqrt(LP1^2-(deltaX(N))^2);
YA2(N)=YA1(N)-deltaY(N); % ratuko A centro Y

```

3.6. Ratuko vertinimas per išorinę laiptelio briauną

Šioje programos dalyje aprašomas platformos ratuko judėjimas vertikalia plokštuma. Žemiau esantis programinio kodo fragmentas naudojamas priekinės platformos ašies, A pusės ratuko vertinimui per išorinę briauną aprašymui. B pusės ratuko aprašymui naudojamas tas pats kodas, tik pakeičiamas kintamųjų pavadinimuose esantis indeksas iš A į B .

```

elseif Flag1 == 3 %
fiA1 = fiA1 + dfA1 ; % naujas kampas A rato sukimui
LA1(N) = LA11 + R*sin(fiA1);
XA1(N)=XA11+R*sin(fiA1)*cos(fAB1); % A centro X
YA1(N)=YA11+R*sin(fiA1)*sin(fAB1); % A centro Y
LC11(N) = LC111; % A centras stovi atsiremes
XC11(N)=XC111 ; % A stovi, jo koordinatės nesikeičia
YC11(N)=YC111 ; % CA stovi, jo koordinatės nesikeičia
HA1(N) = R + CN(LaiptoNr_1,3) - R*(1-cos(fiA)); %
HC11(N) = R + CN(LaiptoNr_1-1,3);
HSN1(N)= CN(LaiptoNr_1-1,3);
Flag1 = 1 ; % pradedama lipti iš naujo
end

```

Žemiau esančioje programinio kodo dalyje pateiktas fragmentas programos, kurioje aprašomas antrojo A ratuko centro koordinatėms nustatymas. Šio ratuko centro koordinatėms naudojamos matematinėje analizė gautos formulės.

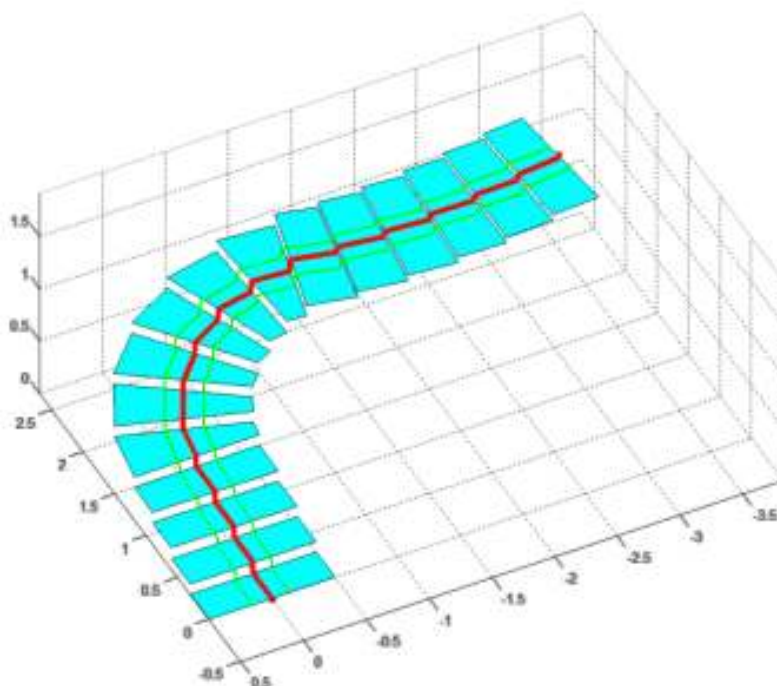
```

LA1(N) = LA11 + R*sin(fiA1);
XP1(N)=CN(NA,1)=CN(NA-1,1)+dx;
YP1(N)=CN(NA,1)=CN(NA-1,1)+dz;
B1(N)=sqrt((XP1-XA1(N))^2-(YP1-XY1(N))^2);
fi31=asin((XA1(N)-XP1)/B1(N));
fi41=acos((R^2+B1(N)^2-LP1^2)/(2*R*B1(N)));
fi51=3,14-fi31-fi41;
YD1(N)=R*sin(fi51);
XD1(N)=R*cos(fi51);
XA1(N)=XA1(N)-XP1-XA1(N)-XD1; % A centro X
YA1(N)=YA1(N)-YP1-YA1(N)-YD1; % A centro Y

```

3.7. Simuliacijos rezultatai

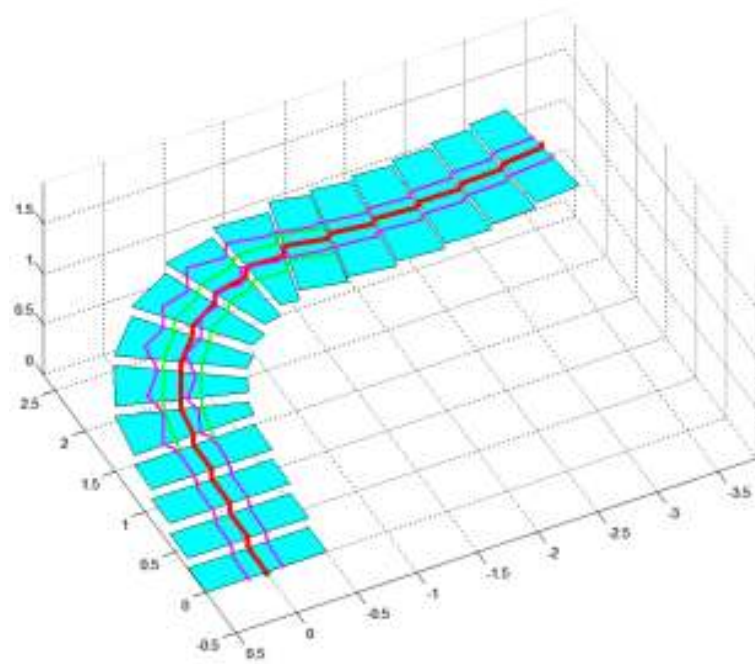
Kompiuterinė platformos simuliacija „MATLAB“ programiniu paketu reikalinga tam, kad įsitikinti matematinės analizės ir sukurto valdymo algoritmo teisingumu. Pirmiausia išbandomas pirmosios ašies judėjimas spiraliniais laiptais tam, kad būtų galima veikiantį metodą ir programos kodą panaudoti sekančių ašių simuliacijai (43 pav.). Laiptų modelyje nubrėžiama vidurio linija, pagal kurią galima lengviau spręsti ar platformos ratų centrų koordinatės juda reikiama kryptimi. Iš pirmosios ašies simuliacijos rezultatų matoma, kad pasirinktas dviejų nepriklausomai varomų ratukų valdymo principas yra tinkamas ir priekinė ašis juda spiraliniais laiptais taip, kaip priklauso.



43 pav. Priekinės ašies judėjimas spiraliniais laiptais

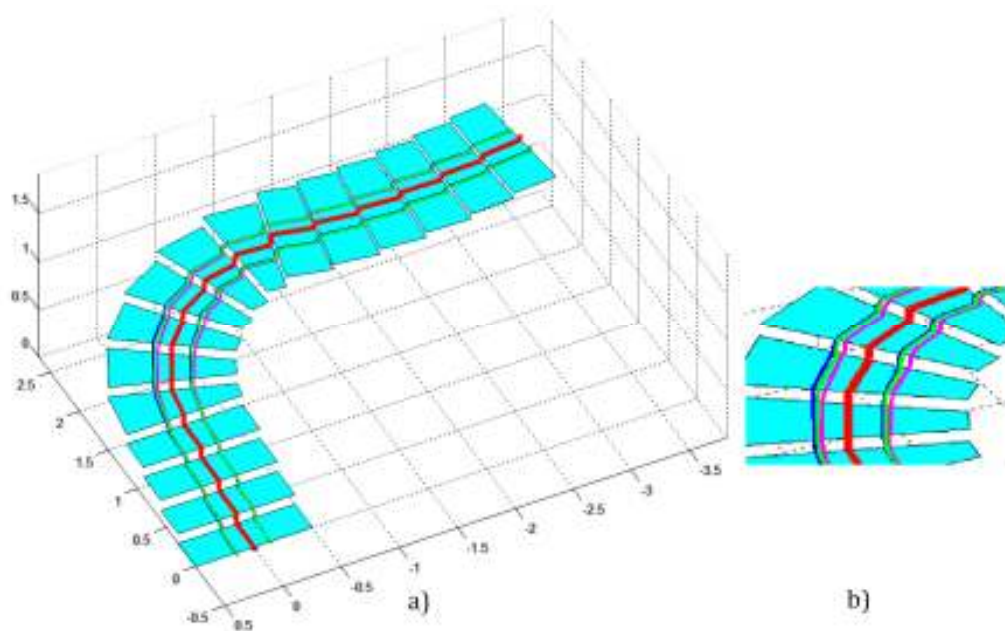
Modeliuojant antrosios ašies lipimą spiraliniais laiptais patikrinama kaip ji judėtų, jeigu būtų tik standžiai sujungta su priekine ašimi, o ratukai nebūtų valdomi atskirai (44 pav.). Tai reiškia, kad antrosios ašies ratukų centrų koordinatės nustatomos tik pagal pirmosios ašies ratukų centrų koordinatės, neįvertinant antrosios ašies ratukų padėties laiptų atžvilgiu.

Įvertinus gautą judėjimo grafiką, daroma prielaida, kad antrosios platformos ratų centrų judėjimo trajektorijų zigzagiškumas atsiranda todėl, kad priekinės ašies ratai, judėdami spiraliniais laiptais, nenukeliauja vienodo atstumo dėl laiptų kreivumo ir pasisukę pakeičia antrosios ašies ratukų centrų koordinatės, nepriklausomai nuo to, kokioje vietoje laiptų atžvilgiu jie yra. Matoma, kad antrosios ašies ratai ne visada juda horizontalioje padėtyje, kartais net yra aukščiau už laiptelio horizontalią plokštumą. Toks platformos judėjimas nėra stabilus ir tenkinantis, todėl nagrinėti kaip nevaldoma judėtų ir trečioji platformos ratų ašis, nėra tikslo. Remiantis gautu grafiku galima įsitikinti, kad ratų sukimosi greičių valdymas pagal laiptų formą yra reikalingas.



44 pav. Antrosios nevaldomos ašies judėjimas spiraliniais laiptais

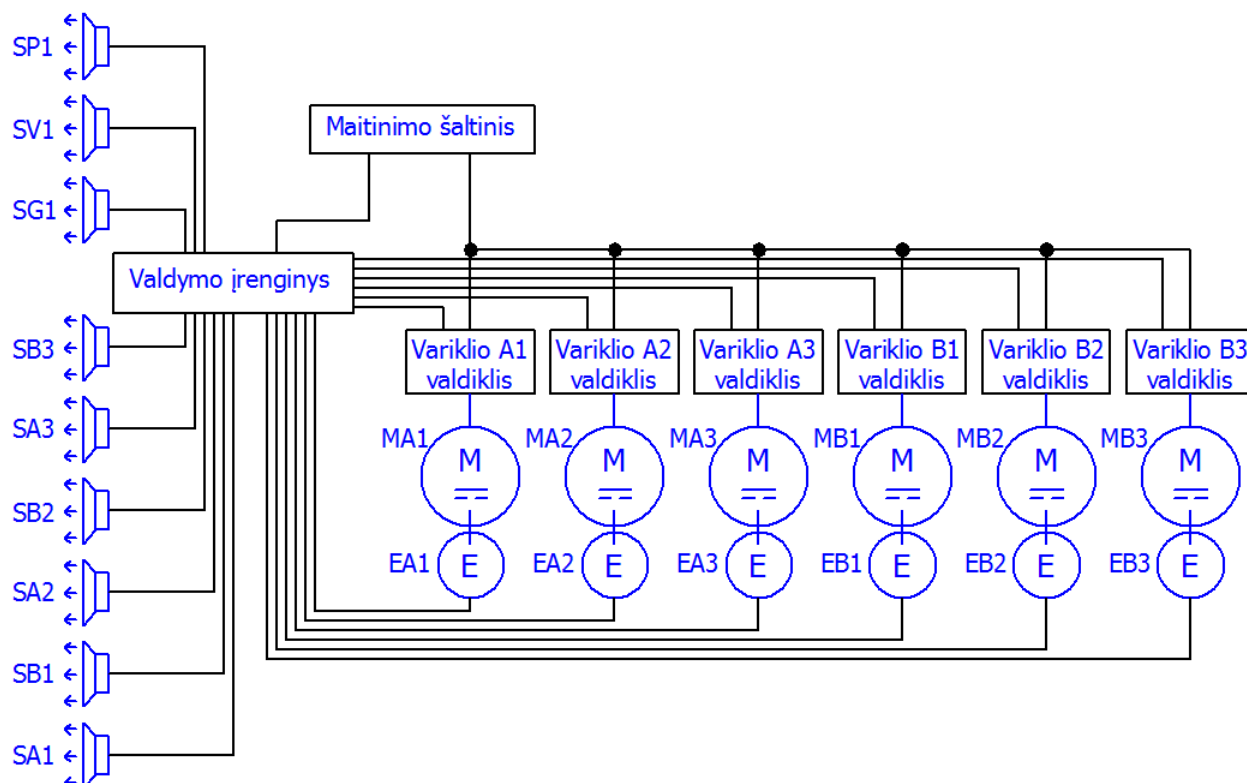
Pasinaudojant matematinėje analizėje atliktais skaičiavimais ir platformos valdymo algoritmu, sumodeliuojamos antrosios ir trečiosios ašių judėjimo trajektorijos (45 pav. a). Kadangi kiekvienas ratas valdomas atskirai, priimta, kad platforma yra lanksti arba jos ratai gali būti vairuojami, todėl visų ratų judėjimo trajektorijos beveik sutampa ir yra beveik lygiagrečios laiptų ašinei linijai. Judėjimo trajektorijų nevienodumą galima matyti padidintame trimačiame grafike (45 pav. b). Taip gali nutikti dėl to, kad antrosios ir trečiosios ašies ratukų centrų koordinatės yra matematinų skaičiavimų rezultatas, o ne duomenų masyvų palyginimas su reikiamaisiais, kaip yra pirmosios ašies atveju. Taip pat dėl to, kad platforma posūkyje juda nevienodu keliu.



45 pav. Trijų valdomų ašių judėjimas spiraliniais laiptais

3.8. Techninė valdymo sistema

Išanalizuotą platformos valdymą pagal laiptų formą techniškai galima realizuoti sistema, kurios funkcinė schema yra pateikta žemiau esančiame paveiksle (46 pav.). Schema skirta platformai, kurioje valdomi tik ratų sukimosi greičiai, o ne jų posūkio kampai. Matoma, kad kiekvienas ratuką varantis elektros variklis ($MA1 - MB3$) yra valdomas nuolatinės srovės elektros valdikliu, kuris keičia variklio maitinimo įtampą pagal gautą valdymo signalą. Variklių velenai sujungti su enkoderiais ($EAI - EB3$), kurie valdymo įrenginiui siunčia grįžtamąjį ryšį apie variklių sukimosi padėtį. Jis reikalingas gauti informacijai ar ratukai pasisuko užduotu kampu. Jeigu ne – atliekama ratuko posūkio korekcija.



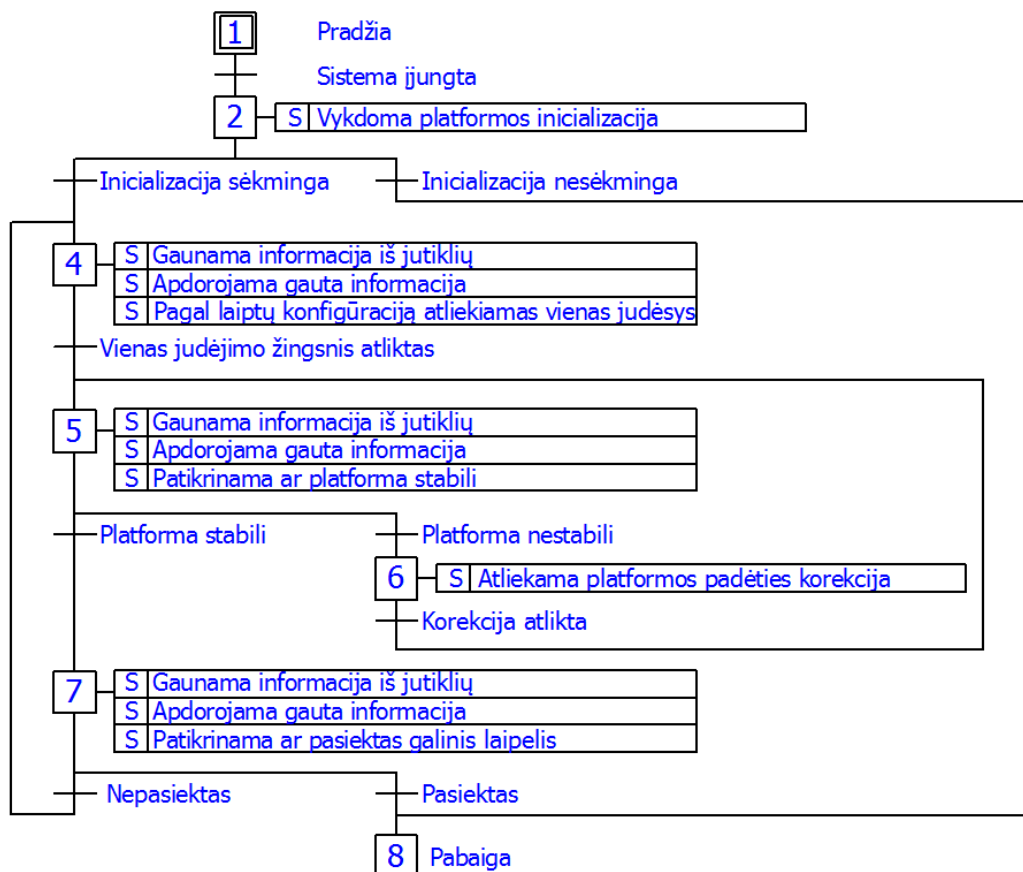
46 pav. Funkcinė valdymo sistemos schema

Signalą apie platformos ratukų padėtį laiptų atžvilgiu kylant siunčia ultragarsiniai jutikliai $SA1 - SB3$, esantys šalia ratukų. Naudojant šiuos jutiklius, nebūtinas pirmosios ašies, kaip vendančiosios, naudojimas, kadangi informaciją apie kiekvieno ratuko padėtį valdymo įrenginys gauna atskirai ir nepriklausomai nuo kitų ratukų. Pagal gautą informaciją valdymo įrenginys nustato ratuko atstumą iki laipto statmens ar išorinės briaunos ir palyginęs gautą atstumą su kitomis gautomis reikšmėmis, valdo platformos ratukų sukimosi greičius taip, kad platforma spiraliniais laiptais judėtų stabiliai. Informacija apie platformos ratų padėtį jai leidžiantis gaunama naudojant ultragarsinius jutiklius $SP1$, $SV1$ ir $SG1$, kurie yra nukreipti į platformos apačią ir nustato atstumą tarp pagrindo ir jutiklio platformos ratukui dar nepasiekus žemesnio laiptelio. Apdorojant

informaciją, gautą iš platformos jutiklių, galima nustatyti ar platforma kyša, ar leidžiasi. Pagal šį judėjimą turi skirtis ir platformos judėjimo valdymas.

3.9. Platformos valdymo algoritmas

Sukurtai platformos techninei valdymo sistemai sudaromas valdymo algoritmas (47 pav.). Šiame algoritme atvaizduojamas apibendrintas platformos valdymas. Inicializacijos, platformos judėjimo, platformos stabilumo ir padėties korekcijos algoritmai gali būti detalizuoti, tačiau tai tikslinga daryti derinant mechaniškai surinktos platformos valdymą. Iš algoritmo matoma, kad platforma juda remdamasi iš atstumo jutiklių gauta ir apdorota informacija. Po kiekvieno atlikto judesio yra labai svarbu patikrinti ar platforma yra stabili. Jeigu ne, tuomet atliekama padėties korekcija, siekiant platformą ant spiralinių laiptų paviršiaus išlaikyti kuo galima stabilesnę.

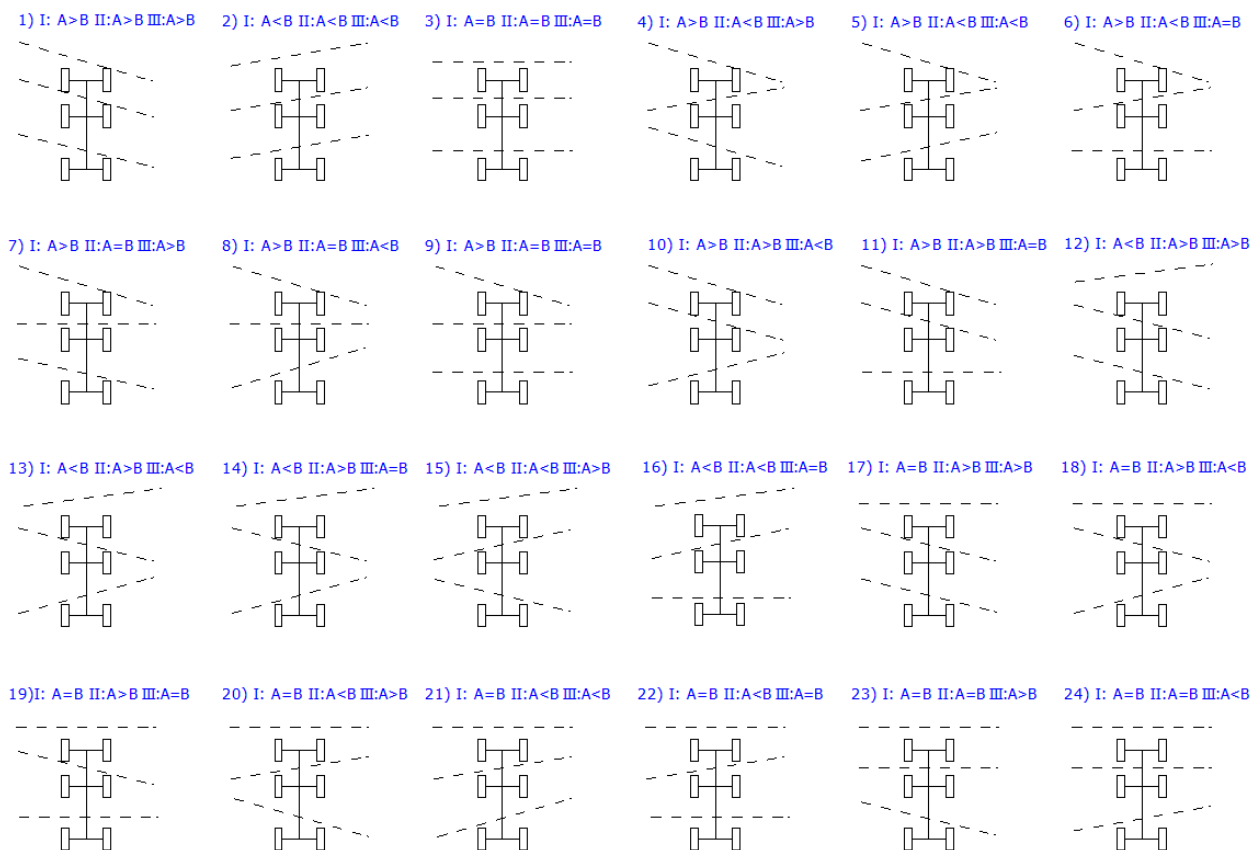


47 pav. Platformos valdymo algoritmas

3.10. Ratų posūkio kampų valdymas pagal ratų ašių atstumus iki laiptų

Valdant platformą pagal atstumą iki laiptų, būtina išsiaiškinti kaip keisti ratų posūkio kampus esant tam tikrai platformos padėčiai spiralinių laiptų atžvilgiu. Iš viso galimi 24-i skirtingi platformos ratų atstumų iki laiptų statmenų deriniai (48 pav.), romėniškais skaitmenimis nurodytos platformos ašys (I – pirmoji, II – vidurinė, III – galinė), A - kairioji platformos pusė, B – dešinioji.

Kokiu kampu turi pasisukti kiekvienas ratukas esant tam tikrai sąlygai, aprašyta žemiau esančioje lentelėje (1 lentelė). Reikia pažymėti, kad surašytos teorinės reikšmės, kurias būtina išbandyti su realia platforma, įvertinant jos bei laiptų matmenis, masės centrą, platformos ir krovinio masę, ir t.t. Taip pat gali būti kombinacijų, kuriose vienas iš variklių turi būti išjungtas, o ratukas judėti laisva eiga. Judėjimo žingsniu pasirinktas vienas laipsnis dėl skaičiavimo paprastumo, tikslesnio platformos judėjimo, tačiau esant mechaniškai sukurtai platformai, jis gali būti kitoks, priklausomai nuo ratus sukančių elektros variklių.



48 pav. Galimos laiptų konfigūracijos

1 lentelė

Platformos ratų posūkio kampai pagal laiptų konfigūraciją

Laiptų konfigūracija		Platformos ratas		Platformos posūkio kampas (laipsniais)			
		A ₁	A ₂	A ₃	B ₁	B ₂	B ₃
1	I: A>B II: A>B III: A>B	2	2	2	1	1	1
2	I: A<B II: A<B III: A<B	1	1	1	1	1	1
3	I: A=B II: A=B III: A=B	1	1	1	1	1	1
4	I: A>B II: A<B III: A>B	2	1	2	1	Išj.	1
5	I: A>B II: A<B III: A<B	2	1	1	1	2	2

6	I: A>B II: A<B III: A=B	2	1	2	1	2	1
7	I: A>B II: A=B III: A>B	2	2	2	1	1	1
8	I: A>B II: A=B III: A<B	2	2	2	1	Išj.	1
9	I: A>B II: A=B III: A=B	2	2	2	1	1	1
10	I: A>B II: A>B III: A<B	2	2	2	1	1	1
11	I: A>B II: A>B III: A=B	2	2	2	1	1	1
12	I: A<B II: A>B III: A>B	1	2	1	2	1	2
13	I: A<B II: A>B III: A<B	1	1	1	2	2	2
14	I: A<B II: A>B III: A=B	1	1	1	2	2	2
15	I: A<B II: A<B III: A>B	1	1	Išj.	2	2	2
16	I: A<B II: A<B III: A=B	1	1	1	2	2	2
17	I: A=B II: A>B III: A>B	2	1	2	2	2	1
18	I: A=B II: A>B III: A<B	2	2	1	2	1	2
19	I: A=B II: A>B III: A=B	2	2	2	2	2	2
20	I: A=B II: A<B III: A>B	2	1	2	2	2	1
21	I: A=B II: A<B III: A<B	2	1	2	2	1	2
22	I: A=B II: A<B III: A=B	2	2	2	2	2	2
23	I: A=B II: A=B III: A>B	2	2	2	2	2	1
24	I: A=B II: A=B III: A<B	2	2	1	2	2	2

3.11. Skyriaus išvados

1. Nevaldant platformos ratų judėjimo greičių, jos judėjimas spiraliniais laiptais tampa nestabilus ir pavojingas kroviniui.
2. Siekiant gauti stabilų platformos judesį spiraliniais laiptais, būtina žinoti kiekvieno rato padėtį laiptų atžvilgiu.
3. Platformos ratų valdymą galima palengvinti žinant galimas laiptų konfigūracijas.

IŠVADOS

1. Kliūčių įveikimui skirtų mobilių platformų tyrimams ir tobulinimui nuolat skiriama daug dėmesio.
2. Iš analizuotų mobilių platformų dėl gerų judėjimo savybių ir konstrukcijos paprastumo pasirinkta „Rocker-Bogie“ tipo platforma, kuri gali būti panaudota kaip paslaugų robotas.
3. Svarbiausias dydis analizuojant paslaugų roboto judėjimą spiraliniais laiptais yra ratų centrų koordinatės, reikalingos ratų padėčiai laiptų atžvilgiu nustatyti.
4. Judant spiraliniais laiptais svarbu įvertinti platformos judėjimo netolygumą ir nevienodą jos ašių lygiagretumą laiptų horizontaliai plokštumai, siekiant išlaikyti kuo galima didesnį platformos stabilumą.
5. Paslaugų roboto judėjimas spiraliniais laiptais valdomas keičiant ratukų judėjimo greitį, priklausomai nuo ratų centrų atstumo iki laipto vertikalios plokštumos arba išorinės briaunos.
6. Priimta, kad platforma gali būti lanksti arba jos ratai vairuojami. Lanksčios platformos atžvilgiu tai reiškia, kad atstumas tarp platformos ratų gali būti kintantis, priklausomai nuo reikiamo judėjimo.
7. Lanksčios arba vairuojamos platformos judėjimas spiraliniais laiptais yra stabilus, todėl tinkamas naudoti kaip paslaugų robotas.

LITERATŪROS SĄRAŠAS

1. I04.hsncdn.com [interaktyvus]. 2016 [žiūrėta 2016-04-25]. Prieiga per internetą: http://i04.hsncdn.com/is/image/HomeShoppingNetwork/prodgrid/folding-cart-with-stair-climbing-wheel-technology-d-2015030616371812~419447_alt2.jpg
2. Topchair.cn [interaktyvus]. 2016 [žiūrėta 2016-04-25]. Prieiga per internetą: <http://www.topchair.cn/blog/upload/201005052034092213.jpg>
3. Quaglia, Giuseppe, Walter Franco and Riccardo Oderio. Wheelchair.q, a motorized wheelchair with stair climbing ability. Mechanism and Machine Theory. 2011, vol. 46, no. 11, pp. 1601-1609.
4. Sinkevičius, Vytenis, Raimondas Klemka and Lina Urbanavičiūtė. The Mathematical Model of Mobile Platform for Static and Dynamic Characteristics Research. Proceedings of 19th International Conference. Mechanics. 2013.
5. KOMSUOĞLU, Haldun. Dynamic Legged Mobility | An Overview. SDU International Journal of Technologic Sciences. 2011, vol. 3, pp. 1-10.
6. Kodlab.seas.upenn.edu [interaktyvus]. 2016 [žiūrėta 2016-04-25]. Prieiga per internetą: <http://kodlab.seas.upenn.edu/uploads/RHex/rhex.jpg>
7. Blogcdn.com [interaktyvus]. 2016 [žiūrėta 2016-04-25]. Prieiga per internetą: <http://www.blogcdn.com/www.engadget.com/media/2013/07/rhex-obstacle-2013-07-25-01.jpg>
8. Hong, Hee Seung, TaeWon Seo, Dongmok Kim, Sunho Kim and Jongwon Kim. Optimal design of hand-carrying rocker-bogie mechanism for stair climbing. Journal of Mechanical Science and Technology. 2013, vol. 27, no. 1, pp. 125-132.
9. Choi, Dongkyu, Jongkyun Oh and Jongwon Kim. Analysis method of climbing stairs with the rocker-bogie mechanism. Journal of Mechanical Science and Technology. 2013, vol. 27, no. 9, pp. 2783-2788.
10. Youtube.com [interaktyvus]. 2016 [žiūrėta 2016-04-25]. Prieiga per internetą: <https://www.youtube.com/watch?v=bP7p5Bd2d50>
11. Thianwiboon, Mongkol and Sangveraphunsiri, Viboon, 2006, Traction Control for a Rocker-Bogie Robot with Wheel-Ground Contact Angle Estimation. Lecture Notes in Computer Science. 2006. Vol. 4020/2006, no. 0302-9743, p. 682-690.
12. Gao, X. Yu. H. and Deng, Z., 2008, The Research on Control of Lunar Rover with Rocker Bogie Based on Bus Network Driving. Information Technology Journal 7. 2008. Vol. 7, no. 1812-5638, p. 1001-1008.

PRIEDAI

1 PRIEDAS. LAIPTŲ MODELIS „MATLAB“ PROGRAMINĖJE APLINKOJE

```
dat = [0 0; 0 1;0 1];
xda=[0,0,0,0];yda=[0,0,0,0];zda=[0,0,0,0];cda=[8,2,2,8];
ptn=[1 3 4 2]; % laiptų plokštumos viršinių eiliškumas brėžimui
spir = [ 0, 4, 0.1, 0.300;
        -1.0, 6, 0.100, 0.300;
        6.0, 6, 0.050, 0.350];
NSmax = size(spir); % laiptų segmentų kiekis
LL = 0.500; % pakopos pusplotis
NA = 1; % NA - absoliutus laipto numeris iš eilės per segmentus
fiN = [0]; % simetrijos linijos kampas N-ojoje pakopoje
CN = [0,0,0]; % X,Y,Z koordinatės ašinės linijos taško N-os pakopos priekyje
CNN = [0,0];
for NS = 1:NSmax % per segmentus, NS - laiptu segmento numeris
    Lw = spir(NS,4); % pakopos gylis
    dz = spir(NS,3); % pakopos aukštis
    df = 0.0; % ašinės linijos pasisukimo per pakopą kampas
    rs = spir(NS,1); % sukimosi spindulys
    if rs ~= 0 % spiralinis segmentas
        df = 2*asin(Lw/(2.0*rs)); % pakopos posūkio kampas
    end
    for N=1:spir(NS,2) % per pakopas, N - laipto numeris segmento viduje
        NA = NA+1; % absoliutus pakopos numeris
        fiN(NA) = fiN(NA-1)+df; % absoliutus ašinės kampas
        CN(NA,1) = CN(NA-1,1)+Lw*cos(fiN(NA-1)+df/2.0); % X ašinės
        CN(NA,2) = CN(NA-1,2)+Lw*sin(fiN(NA-1)+df/2.0); % Y ašinės
        CN(NA,3) = CN(NA-1,3)+dz; % Z ašinės absoliutus laipto aukštis
        CNN(NA,1)= N; CNN(NA,2)=N; % numerio išimimas
    end
end
end
NAmax=NA; % maksimalus pakopų kiekis
maxX = max(CN(:,1)); minX = min(CN(:,1));
maxY = max(CN(:,2)); minY = min(CN(:,2));
maxZ = max(CN(:,3)); minZ = min(CN(:,3));
h4 = axes('position', [ .1 0.1 .8 .8]);
axis([minX-LL maxX+LL minY-LL maxY+LL minZ maxZ+0.6]) % X,Y,Z asių ribos
axis equal % ('square') equal
view([-1,0.5,1]); % view([-1,0.5,1][0,0,1])
grid on
plate = patch(xda,yda,zda,cda,...
    'Marker','o','MarkerFaceColor','flat','FaceColor','none');
Ne = NAmax-1;
for N=1:Ne % koordinatės laipto plokštumoms ir jų brėžimas
    xda(1)=CN(N,1)-LL*cos(pi/2-fiN(N)); % kairės pusės priekis
    xda(2)=CN(N,1)+LL*cos(pi/2-fiN(N)); % dešinės pusės priekis
    xda(3)=CN(N+1,1)-LL*cos(pi/2-fiN(N+1)); % kairės pusės galas
    xda(4)=CN(N+1,1)+LL*cos(pi/2-fiN(N+1)); % dešinės pusės galas
    yda(1)=CN(N,2)+LL*sin(pi/2-fiN(N)); % kairės pusės priekis
    yda(2)=CN(N,2)-LL*sin(pi/2-fiN(N)); % dešinės pusės priekis
    yda(3)=CN(N+1,2)+LL*sin(pi/2-fiN(N+1));
    yda(4)=CN(N+1,2)-LL*sin(pi/2-fiN(N+1));
    zda(1)=CN(N,3);zda(2)=CN(N,3);zda(3)=CN(N,3);zda(4)=CN(N,3);
    plate = patch(xda(:),yda(:),zda(:),cda(:),'Marker','none',...
        'MarkerFaceColor','flat','FaceColor','c','Faces',ptn,...
        'EraseMode','none');
end
end
for N=1:Ne % koordinatės ašinei linijai ir jos brėžimas
    dat(1,1)=CN(N,1);dat(1,2)=CN(N+1,1); % horizontali plokštuma
    dat(2,1)=CN(N,2);dat(2,2)=CN(N+1,2);
    dat(3,1)=CN(N,3);dat(3,2)=CN(N,3);
    % laiptų centro linijos brėžimas
    % set(bar0 , 'xdata', dat(1,:), 'ydata', dat(2,:), 'zdata', dat(3,:));
```

```
drawnow
dat(1,1)=CN(N+1,1);dat(1,2)=CN(N+1,1); % vertikali plokštuma
dat(2,1)=CN(N+1,2);dat(2,2)=CN(N+1,2);
dat(3,1)=CN(N,3);dat(3,2)=CN(N+1,3);
drawnow
end
```

2 PRIEDAS. PLATFORMOS APRAŠYMAS IR JUDĖJIMAS LAIPTAIS

```
% ----- platformos judėjimas laiptais
R=0.1; % įvedamas ratuko spindulys
Ha=0.25; % įvedamas pirmojo ir antrojo ratuko jungties ilgis
Hb=0.5; % įvedamas trečiojo ratuko jungties ilgis
LP1=0.25; % įvedamas atstumas tarp pirmojo ir antrojo ratuko
Ha1=sqrt(Ha^2-(LP1/2)^2); % apskaičiuojamas atstumas pagal Y tarp taško H ir
ratukų centro
LP2=sqrt(Hb^2-Ha1^2)-LP1/2;% randamas atstumas tarp antrojo ir trečiojo ratuko
LaiptuKiekis = Nmax-1; % laiptų kiekis
L1M = CN(:,1); % tikrinama ar ratukas pasiekė galinę padėtį
LAB1 = 1 ; % atstumas tarp ratų, platformos plotis
LAB2 = 1 ; % atstumas tarp ratų, platformos plotis
LAB3 = 1 ; % atstumas tarp ratų, platformos plotis
% kintamasis pirmojo ratuko 1, antrojo ratuko 2 ir trečiojo ratuko 3
LaiptoNr_1 = 1 ; LaiptoNr_1 = 1 ; LaiptoNr_1 = 1; % laipto numeris trims
ratukams
Flag1 = 1 ; Flag2 = 1 ; Flag3 = 1 ;% FLAG - būsenos vėliavėlė
% 1 - ratukas juda horizontalia kryptimi
% 11 - laukiama, kol kitas ratas pririedės iki atramos
% 2 - ratukas juda vertikalia kryptimi
% 3 - ratukas verčiasi per laipto išorinę briauną
% 4 - pabaiga
fi1 = 0.0 ; fi2 = 0.0 ; fi3 = 0.0 ;% ratukų sukimosi kampai
df1 = 0.0025 ; df2 = 0.00025; df3 = 0.00025; % ratukų kampo žingsnis, gali būti
skirtingi
% ----- masyvai koodinatems
Nmax = 1000000; % skaiciavimo žingsniu kiekis
L1(1:Nmax) = 0.0; L2(1:Nmax)= 0.0; L3(1:Nmax)= 0.0; % Ratuko centro padėtis X
kryptimi
H1(1:Nmax) = 0.0; H2(1:Nmax)= 0.0; H3(1:Nmax)= 0.0; % Ratuko centro padėtis Y
kryptimi
% AB kampas ir atstumo tarp AB ir (LA0-LB0) skirtumas
fiAB1(1:Nmax) = 0.0; deltaAB1(1:Nmax)= 0.0; HSN1(1:Nmax)=0.0;
fiAB2(1:Nmax) = 0.0; deltaAB2(1:Nmax)= 0.0; HSN2(1:Nmax)=0.0;
fiAB3(1:Nmax) = 0.0; deltaAB3(1:Nmax)= 0.0; HSN3(1:Nmax)=0.0;
% ----- pradiniai dydžiai masyvuose, aprašantys priekinę ašį-----
% A ratai kairėje pusėje, B ratai dešinėje pusėje
% pradinėje padėtyje nurodoma rato A padėtis pagal X ir Y ir kampą su
% laipto briauna laipsniais, 0 - stovi lygiagrečiai
N = 1 ; % indeksas masyvui
fABAdeg = 0.0 ; % platformos kampas atžvilgiu pirmo laipto briaunos
% kampas fAB keičiasi priklausomai nuo algoritmo, RS ar dfB
fAB1 = fAB1deg*pi/180.0 ; % kampas fAB, išreikštas radianais
LA10 = -0.5 ; LB10 = -0.5 ; % A,B mazgo atstumas nuo pradžios kiekvienam ciklui
LA1(N) = LA0 ; L1B(N) = LB10 ; % A,B padėtys X kryptimi, judama X kryptimi
HA1(N) = R ; HB1(N) = R ; % A, B padėtis Y kryptimi
% pradinėje padėtyje B mazgas gali būti arčiau ar toliau laipto
XA1(N) = LA10; XB1(N) = XA1(N)+LAB1*sin(fAB1);
YA1(N) = -LAB1/2.0; YB1(N) = YA1(N)+LAB1*cos(fAB1);
% aprašoma pirmojo ratuko padėtis
LR1(N) = 0; % padėtis X kryptimi
HR1(N) = R; % padėtis Y kryptimi
XR1(N) = XA1(N); % padėtis X kryptimi
YR1(N) = YA1(N);% padėtis X kryptimi
deltaAB1(N)= 0.0; fiAB1(N) = 0.0; %LA(N)-LB(N);
LB100 = LB10; fiD1A1=0.0 ;
% ----- pradiniai dydžiai masyvuose, aprašantys vidurinę ašį-----
% A ratai kairėje pusėje, B ratai dešinėje pusėje
% pradinėje padėtyje nurodoma rato A padėtis pagal X ir Y ir kampą su
```

```

% laipto briauna laipsniais, 0 - stovi lygiagrečiai
N = 1 ; % indeksas masyvui
fAB2deg = 0.0 ; % platformos kampas atžvilgiu pirmo laipto briaunos
% kampas fAB keičiasi priklausomai nuo algoritmo, RS ar dfB
fAB2 = fAB2deg*pi/180.0 ; % kampas fAB, išreikštas radianais
LA20 = -0.5+LP1 ; LB20 = -0.5+LP1 ; % A,B mazgo atstumas nuo pradžios kiekvienam
ciklui
LA2(N) = LA20 ; L1B(N) = LB20 ; % A,B padėtys X kryptimi, judama X kryptimi
HA2(N) = R ; HB2(N) = R ; % A, B padėtis Y kryptimi
% pradinėje padėtyje B mazgas gali būti arčiau ar toliau laipto
XA2(N) = LA20; XB2(N) = XA2(N)+LAB2*sin(fAB2);
YA2(N) = -LAB2/2.0; YB2(N) = YA2(N)+LAB2*cos(fAB2);
% aprašoma pirmojo ratuko padėtis
LR2(N) = 0; % padėtis X kryptimi
HR2(N) = R; % padėtis Y kryptimi
XR2(N) = XA2(N); % padėtis X kryptimi
YR2(N) = YA2(N); % padėtis X kryptimi
deltaAB2(N)= 0.0; fiAB2(N) = 0.0; %LA(N)-LB(N);
LB200 = LB20; fiD1A2=0.0 ;
% ----- pradiniai dydžiai masyvuose, aprašantys galinę ašį-----
% A ratai kairėje pusėje, B ratai dešinėje pusėje
% pradinėje padėtyje nurodoma rato A padėtis pagal X ir Y ir kampą su
% laipto briauna laipsniais, 0 - stovi lygiagrečiai
N = 1 ; % indeksas masyvui
fAB3deg = 0.0 ; % platformos kampas atžvilgiu pirmo laipto briaunos
% kampas fAB keičiasi priklausomai nuo algoritmo, RS ar dfB
fAB3 = fAB3deg*pi/180.0 ; % kampas fAB, išreikštas radianais
LA30 = -0.5+LP1+LP2 ; LB30 = -0.5+LP1+LP2 ; % A,B mazgo atstumas nuo pradžios
kiekvienam ciklui
LA3(N) = LA30 ; L1B(N) = LB30 ; % A,B padėtys X kryptimi, judama X kryptimi
HA3(N) = R ; HB3(N) = R ; % A, B padėtis Y kryptimi
% pradinėje padėtyje B mazgas gali būti arčiau ar toliau laipto
XA3(N) = LA30; XB3(N) = XA3(N)+LAB3*sin(fAB3);
YA3(N) = -LAB3/2.0; YB3(N) = YA3(N)+LAB3*cos(fAB3);
% aprašoma pirmojo ratuko padėtis
LR3(N) = 0; % padėtis X kryptimi
HR3(N) = R; % padėtis Y kryptimi
XR3(N) = XA3(N); % padėtis X kryptimi
YR3(N) = YA3(N); % padėtis X kryptimi
deltaAB3(N)= 0.0; fiAB3(N) = 0.0; %LA(N)-LB(N);
LB300 = LB30; fiD1A3=0.0 ;

```


3 PRIEDAS. A PUSĖS RATUKŲ JUDĖJIMAS SPIRALINIAIS LAIPTAIS

```
N = 2 ; % sekantis taškas nuo starto pozicijos
while (FlagA<4) % patikrinama ar kylama, ar leidžiamasi
    if CN(LaiptoNr_1,3) > CN(LaiptoNr_1+1,3) % patikrinama ar kylama
        LstopA1 = R/4.0; % sustoti atstumu nuo laipto kai leidžiamasi
    else
        LstopA1 = R; % sustoti atstumu nuo laipto kai kylama
    end
    fAB1=atan(XA1(N-1)-XB1(N-1))/(YA1(N-1)-YB1(N-1)); % kampas platformos
    fAB=0;
% pirmojo ratuko riedėjimas horizontaliai
    if Flag1 == 1 % riedėjimas horizontaliai
        fiA1 = fiA1 + dfA1 ; % naujas kampas sukantis aplink ratuko ašį
        LA1(N) = fiA1*R + LA10 ; % A ratuko centro padėtis
        fAB1=atan((XA1(N-1)-XB1(N-1))/(YA1(N-1)-YB1(N-1))); % platformos kampas
        fiDlA1=dfA1*R/LAB1; % posūkio kampas aplink B
        fiLL1=fAB1-fiDlA1;
        XA1(N)=XB1(N-1)-LAB1*sin(fiLL1); % ratuko A centro X
        YA1(N)=YB1(N-1)-LAB1*cos(fiLL1); % ratuko A centro Y
        LC11(N) = LA1(N); % ratuko centras juda kartu su A tasku
        XC11(N)=XA1(N); % ratuko centras apačioje juda kartu su A tasku
        YC11(N)=YA1(N); % ratuko centras apačioje juda kartu su A tasku
        HA11(N) = R + CN(LaiptoNr_1,3); % mazgo A aukstis
        HC11(N) = R + CN(LaiptoNr_1,3); % ratuko centro aukstis
        HSN1(N)=CN(LaiptoNr_1,3);
        if XC11(N) >= CN(LaiptoNr_1+1,1) - LstopA1 % riba, kai ratukas atsiremia
            if LaiptoNr_1 < LaiptuKiekis
                LaiptoNr_1 = LaiptoNr_1 + 1 ;
            else
                Flag1=4;
            end
            LA11 = LA1(N); % tasko A padėtis prieš sukantis
            XA11=XA1(N); % atsimenama padėtis prieš sukantis
            YA11=YA1(N);
            LC111 = LC11(N); % rato centro padėtis prieš sukantis ant laipto
            XC111=XC11(N);
            YC111=YC11(N);
            fiA1 = 0.0 ; %
            Flag1 = 11; % laukiama, kol kita pusė privažiuos
        end % pabaiga XA(N) >= L1M(LaiptoNr_1+1) - LstopA
    elseif Flag1 == 11 % laukiame B iki atramos, stovime
        XA1(N)=XA1(N-1);
        YA1(N)=YA1(N-1);
        XC11(N)=XC11(N-1);
        YC11(N)=YC11(N-1);
        LC11(N) = LC11(N-1);
        LA1(N) = LA1(N-1);
% pirmojo ratuko judėjimas vertikaliai
    if Flag1 == 2 % judėjimas
        fiA1 = fiA1 + dfA1 ; % naujas kampas sukantis aplink ratuko ašį
        LA1(N) = fiA1*R + LA10 ; % A ratuko centro padėtis
        fAB1=atan((XA1(N-1)-XB1(N-1))/(YA1(N-1)-YB1(N-1))); % platformos kampas
        fiDlA1=dfA1*R/LAB1; % posūkio kampas aplink B
        fiLL1=fAB1-fiDlA1;
        XA1(N)=XB1(N-1)-LAB1*cos(fiLL1); % ratuko A centro X
        YA1(N)=YB1(N-1)-LAB1*sin(fiLL1); % ratuko A centro Y
        LC11(N) = LA1(N); % ratuko centras juda kartu su A tasku
        XC11(N)=XA1(N); % ratuko centras apačioje juda kartu su A tasku
        YC11(N)=YA1(N); % ratuko centras apačioje juda kartu su A tasku
        HA11(N) = R + CN(LaiptoNr_1,3); % mazgo A aukstis
```

```

HC11(N) = R + CN(LaiptoNr_1,3); % ratuko centro aukstis
HSN1(N)= CN(LaiptoNr_1,3);
if XC11(N) >= CN(LaiptoNr_1+1,1) - LstopA1 % riba, kai ratukas atsiremia
    if LaiptoNr_1 < LaiptuKiekis
        LaiptoNr_1 = LaiptoNr_1 + 1 ;
    else
        Flag1=4;
    end
    LA11 = LA1(N); % tasko A padetis pries sukantis
    XA11=XA1(N); % atsimenama padetis pries sukantis
    YA11=YA1(N);
    LC111 = LC11(N); % rato centro padetis pries sukantis ant laipto
    XC111=XC11(N);
    YC111=YC11(N);
    fiA1 = 0.0 ; %
    Flag1 = 11; % laukiama, kol kita puse privažiuos
end % pabaiga XA(N) >= L1M(LaiptoNr_1+1) - LstopA
elseif Flag1 == 11 % laukiame B iki atramos, stovime
    XA1(N)=XA1(N-1);
    YA1(N)=YA1(N-1);
    XC11(N)=XC11(N-1);
    YC11(N)=YC11(N-1);
    LC11(N) = LC11(N-1);
    LA1(N) = LA1(N-1);
% pirmojo ratuko judėjimas vertimasis
elseif Flag1 == 3 %
fiA1 = fiA1 + dfA1 ; % naujas kampas A mazgo sukimui, nuo 0
LA1(N) = LA11 + R*sin(fiA1);
    XA1(N)=XA11+R*sin(fiA1)*cos(fAB1); % A centro X
    YA1(N)=YA11+R*sin(fiA1)*sin(fAB1); % A centro Y
LC11(N) = LC111; % C1 centras stovi atsiremes
    XC11(N)=XC111 ; % C1 stovi, jo koordinates nesikeicia
    YC11(N)=YC111 ; % C1 stovi, jo koordinates nesikeicia
HA1(N) = R + CN(LaiptoNr_1,3) - R*(1-cos(fiA)); %
HC11(N) = R + CN(LaiptoNr_1-1,3);
HSN1(N)= CN(LaiptoNr_1-1,3);
Flag1 = 1 ; % pradedama lipti iš naujo
end
% antrojo ratuko riedėjimas horizontaliai
if Flag2 == 1 % riedėjimas horizontaliai
    fiA2 = fiA2 + dfA2 ; % naujas kampas sukantis aplink ratuko ašį
    LA2(N) = fiA2*R + LA20 ; % A ratuko centro padetis
    fAB2=atan((XA2(N-1)-XB2(N-1))/(YA2(N-1)-YB2(N-1))); % platformos kampas
    fiD1A2=dfA2*R/LAB2; % posūkio kampas aplink B
    fiLL2=fAB2-fiD1A2;
    YP1(N)=CN(NA,1)=CN(NA-1,1)+dz;
    YA2(N)=YP1+R; % ratuko A centro Y
    deltaY(N)=YA11-(YP1(N)+R);
    deltaX(N)=sqrt(LP1^2-(deltaY(N))^2);
    XA2(N)=YA1(N)-deltaX(N); % ratuko A centro X
    HA12(N) = R + CN(LaiptoNr_2,3); % mazgo A aukstis
    HC12(N) = R + CN(LaiptoNr_2,3); % ratuko centro aukstis
    HSN2(N)= CN(LaiptoNr_2,3);
    if XC12(N) >= CN(LaiptoNr_2+1,1) - LstopA2 % riba, kai ratukas atsiremia
        if LaiptoNr_2 < LaiptuKiekis
            LaiptoNr_2 = LaiptoNr_2 + 1 ;
        else
            Flag2=4;
        end
        LA12 = LA2(N); % tasko A padetis pries sukantis
        XA12=XA2(N); % atsimenama padetis pries sukantis
        YA12=YA2(N);
        LC112 = LC12(N); % rato centro padetis pries sukantis ant laipto
        XC112=XC12(N);

```

```

        YC112=YC12(N);
        fiA2 = 0.0 ; %
        Flag2 = 11; % laukiama, kol kita pusė privažiuos
    end % pabaiga XA(N) >= L1M(LaiptoNr_2+1) - LstopA
elseif Flag2 == 11 % laukiame B iki atramos, stovime
    XA2(N)=XA2(N-1);
    YA2(N)=YA2(N-1);
    XC12(N)=XC12(N-1);
    YC12(N)=YC12(N-1);
    LC12(N) = LC12(N-1);
    LA2(N) = LA2(N-1);
% antrojo ratuko riedėjimas vertikalčiai
    if Flag2 == 2 % judėjimas
        fiA2 = fiA2 + dfA2 ; % naujas kampas sukantis aplink ratuko ašį
        LA2(N) = fiA2*R + LA20 ; % A ratuko centro padėtis
        fAB2=atan((XA2(N-1)-XB2(N-1))/(YA2(N-1)-YB2(N-1))); % platformos kampas
        fiD1A2=dfA2*R/LAB2; % posūkio kampas aplink B
        fiLL2=fAB2-fiD1A2;
        XA2(N)=XB2(N-1)-LAB2*cos(fiLL2); % ratuko A centro X
        YA2(N)=YB2(N-1)-LAB2*sin(fiLL2); % ratuko A centro Y
        LC12(N) = LA2(N); % ratuko centras juda kartu su A tasku
        XC12(N)=XA2(N); % ratuko centras apačioje juda kartu su A tasku
        YC12(N)=YA2(N); % ratuko centras apačioje juda kartu su A tasku
        HA12(N) = R + CN(LaiptoNr_2,3); % mazgo A aukstis
        HC12(N) = R + CN(LaiptoNr_2,3); % ratuko centro aukstis
        HSN2(N)= CN(LaiptoNr_2,3);
        if XC12(N) >= CN(LaiptoNr_2+1,1) - LstopA2 % riba, kai ratukas atsiremia
            if LaiptoNr_2 < LaiptuKiekis
                LaiptoNr_2 = LaiptoNr_2 + 1 ;
            else
                Flag1=4;
            end
            LA12 = LA2(N); % tasko A padėtis prieš sukantis
            XA12=XA2(N); % atsimenama padėtis prieš sukantis
            YA12=YA2(N);
            LC112 = LC12(N); % rato centro padėtis prieš sukantis ant laipto
            XC112=XC12(N);
            YC112=YC12(N);
            fiA2 = 0.0 ; %
            Flag2 = 11; % laukiama, kol kita pusė privažiuos
        end % pabaiga XA(N) >= L1M(LaiptoNr_1+1) - LstopA
    elseif Flag2 == 11 % laukiame B iki atramos, stovime
        XA2(N)=XA2(N-1);
        YA2(N)=YA2(N-1);
        XC12(N)=XC12(N-1);
        YC12(N)=YC12(N-1);
        LC12(N) = LC12(N-1);
        LA2(N) = LA2(N-1);
% antrojo ratuko vertimasis
    elseif Flag2 == 3 % sukames aplink ašį tol, kol atsiguls ant laipto
        fiA1 = fiA1 + dfA1 ; % naujas kampas A mazgo sukimui, nuo 0
        LA1(N) = LA11 + R*sin(fiA1);
        XP1(N)=CN(NA,1)=CN(NA-1,1)+dx;
        YP1(N)=CN(NA,1)=CN(NA-1,1)+dz;
        B1(N)=sqrt((XP1-XA1(N))^2-(YP1-XY1(N))^2);
        fi31=asin((XA1(N)-XP1)/B1(N));
        fi41=acos((R^2+B1(N)^2-LP1^2)/(2*R*B1(N)));
        fi51=3,14-fi31-fi41;
        YD1(N)=R*sin(fi51);
        XD1=R*cos(fi51);
        XA1(N)=XA1(N)-XP1-XA1(N)-XD1;% A centro X
        YA1(N)=YA1(N)-YP1-YA1(N)-YD1; % A centro Y
        LC11(N) = LC111; % centras stovi atsiremes
        XC11(N)=XC111 ;

```

```

        YC11(N)=YC111 ;
        HA11(N) = R + CN(LaiptoNr_1,3) - R*(1-cos(fiA1)); %
        HC11(N) = R + CN(LaiptoNr_1-1,3);
        HSN1(N)= CN(LaiptoNr_1-1,3);
        Flag2 = 1 ; % pradedama lipti iš naujo
    end
% šarnyro H aprašymas
    if Flag1 == 3
        fiHA = fiHA + dfHA ; % naujas kampas sukantis aplink ratuko ašį
        LAHA(N) = fiHA*R + LAHA ; % A ratuko centro padėtis
        fHA=atan((XHA(N-1)-XHA(N-1))/(YHA(N-1)-YHA(N-1))); % platformos kampas
        fiD1HA=dfHA*R/LAHA; % posūkio kampas aplink B
        fi1HA(N)=acos((XA1(N)-XA2(N))/LP1);
        fiBA=acos((2Ha*LP1)/((-Ha)^2+Ha^2+LP1^2));
        fi2HA(N)=3,14-fi1HA(N)-fiBA;
        XHA=HH(N)*cos(fi2HA(N)); % šarnyro X koordinatė
        YHA=Ha*sin(fi2HA(N)); % šarnyro Y koordinatė
% trečiojo ratuko riedėjimas horizontaliai
    if Flag3 == 1 % riedėjimas horizontaliai
        fiA3 = fiA3 + dfA3 ; % naujas kampas sukantis aplink ratuko ašį
        LA3(N) = fiA3*R + LA30 ; % A ratuko centro padėtis
        fAB3=atan((XHA(N-1)-XHA(N-1))/(YHA(N-1)-YHA(N-1))); % platformos kampas
        fiD1A3=dfA3*R/LAB3; % posūkio kampas aplink B
        fiLL3=fAB3-fiD1A3;
        YP3(N)=CN(NA,1)=CN(NA-1,1)+dz;
        YA3(N)=YP3+R; % ratuko A centro Y
        deltaY(N)=YHA-(YP3(N)+R);
        deltaX(N)=sqrt(Hb^2-(deltaY(N))^2);
        XA3(N)=YHA(N)-deltaX(N); % ratuko A centro X
        HA13(N) = R + CN(LaiptoNr_3,3); % mazgo A aukstis
        HC13(N) = R + CN(LaiptoNr_3,3); % ratuko centro aukstis
        HSN3(N) = CN(LaiptoNr_3,3);
        if XC13(N) >= CN(LaiptoNr_3+1,1) - LstopA3 % riba, kai ratukas atsiremia
            if LaiptoNr_3 < LaiptuKiekis
                LaiptoNr_3 = LaiptoNr_3 + 1 ;
            else
                Flag3=4;
            end
            LA13 = LA3(N); % tasko A padėtis prieš sukantis
            XA13=XA3(N); % atsimenama padėtis prieš sukantis
            YA13=YA3(N);
            LC113 = LC13(N); % rato centro padėtis prieš sukantis ant laipto
            XC113=XC13(N);
            YC113=YC13(N);
            fiA3 = 0.0 ; %
            Flag3 = 11; % laukiama, kol kita pusė privažiuos
        end % pabaiga XA(N) >= L1M(LaiptoNr_3+1) - LstopA
    elseif Flag3 == 11 % laukiame B iki atramos, stovime
        XA3(N)=XA3(N-1);
        YA3(N)=YA3(N-1);
        XC13(N)=XC13(N-1);
        YC13(N)=YC13(N-1);
        LC13(N) = LC13(N-1);
        LA3(N) = LA3(N-1);
% trečiojo ratuko judėjimas vertikaliai
    if Flag3 == 2 % judėjimas
        fiA3 = fiA3 + dfA3 ; % naujas kampas sukantis aplink ratuko ašį
        LA3(N) = fiA3*R + LA30 ; % A ratuko centro padėtis
        fAB3=atan((XHA(N-1)-XHA(N-1))/(YHA(N-1)-YHA(N-1))); % platformos kampas
        fiD1A3=dfA3*R/LAB3; % posūkio kampas aplink B
        fiLL3=fAB3-fiD1A3;
        XA3(N)=XB3(N-1)-LAB3*cos(fiLL3); % ratuko A centro X
        YA3(N)=YB3(N-1)-LAB3*sin(fiLL3); % ratuko A centro Y
        LC13(N) = LA3(N); % ratuko centras juda kartu su A tasku
    end

```

```

XC13(N)=XA3(N); % ratuko centras apačioje juda kartu su A tasku
YC13(N)=YA3(N); % ratuko centras apačioje juda kartu su A tasku
HA13(N) = R + CN(LaiptoNr_3,3); % mazgo A aukstis
HC13(N) = R + CN(LaiptoNr_3,3); % ratuko centro aukstis
HSN3(N)= CN(LaiptoNr_3,3);
if XC13(N) >= CN(LaiptoNr_3+1,1) - LstopA3 % riba, kai ratukas atsiremia
    if LaiptoNr_3 < LaiptuKiekis
        LaiptoNr_3 = LaiptoNr_2 + 1 ;
    else
        Flag3=4;
    end
    LA13 = LA3(N); % tasko A padėtis prieš sukantis
    XA13=XA3(N); % atsimenama padėtis prieš sukantis
    YA13=YA3(N);
    LC113 = LC13(N); % rato centro padėtis prieš sukantis ant laipto
    XC113=XC13(N);
    YC113=YC13(N);
    fiA3 = 0.0 ; %
    Flag3 = 11; % laukiama, kol kita pusė privažiuos
end % pabaiga XA(N) >= L1M(LaiptoNr_1+1) - LstopA
elseif Flag3 == 11 % laukiame B iki atramos, stovime
    XA3(N)=XA3(N-1);
    YA3(N)=YA3(N-1);
    XC13(N)=XC13(N-1);
    YC13(N)=YC13(N-1);
    LC13(N) = LC13(N-1);
    LA3(N) = LA3(N-1);
% trečiojo ratuko vertimasis
elseif Flag == 3 % sukames aplink ašį tol, kol atsiguls ant laipto
    fiA1 = fiA1 + dfA1 ; % naujas kampas A mazgo sukimui, nuo 0
    LA1(N) = LA11 + R*sin(fiA1);
    XP1(N)=CN(NA,1)=CN(NA-1,1)+dx;
    YP1(N)=CN(NA,1)=CN(NA-1,1)+dz;
    B1(N)=sqrt((XP1-XA1(N))^2-(YP1-XY1(N))^2);
    fi31=asin((XA1(N)-XP1)/B1(N));
    fi41=acos((R^2+B1(N)^2-LP1^2)/(2*R*B1(N)));
    fi51=3,14-fi31-fi41;
    YD1(N)=R*sin(fi51);
    XD1=R*cos(fi51);
    XA1(N)=XA1(N)-XP1-XA1(N)-XD1;% A centro X
    YA1(N)=YA1(N)-YP1-YA1(N)-YD1; % A centro Y
    LC11(N) = LC111; % centras stovi atsiremes
        XC11(N)=XC111 ;
        YC11(N)=YC111 ;
    HA11(N) = R + CN(LaiptoNr_1,3) - R*(1-cos(fiA1)); %
    HC11(N) = R + CN(LaiptoNr_1-1,3);
    HSN1(N)= CN(LaiptoNr_1-1,3);
    Flag2 = 1 ; % pradedama lipti iš naujo
end

```