

INFORMATIKOS FAKULTETAS

[**T125B158 Robotų programavimo technologijos**](https://moodle.ktu.edu/course/view.php?id=6144)

**Projektas**

Studentai:

Eligijus Kiudys IFF-7/14

Martynas Girdžiūna IFF-7/14

Simonas Brasas IFF-7/9

Dėstytojai:

Doc. Narbutaitė Lina

Doc. Brūzgienė Rasa

Doc. Prarkt. Laurutis Remigijus

KAUNAS 2020

Turinys

[1. Projektinio darbo užduotis 4](#_Toc57663099)

[2. Užduoties ir reikalavimų analizė 4](#_Toc57663100)

[3. Roboto valdymo scenarijus 5](#_Toc57663101)

[3.1. Linijos sekimo scenarijus 5](#_Toc57663102)

[3.1.1. Dešinės pusės sensorius nebemato linijos 5](#_Toc57663103)

[3.1.2. Kairės pusės sensorius nebemato linijos 5](#_Toc57663104)

[3.1.3. Vidurinis sensorius mato liniją 5](#_Toc57663105)

[3.1.4. Visi sensoriai linijos nebeaptinka ir paskutinis sensorius kuris aptiko liniją buvo kairės pusės sensorius 5](#_Toc57663106)

[3.1.5. Visi sensoriai linijos nebeaptinka ir paskutinis sensorius kuris aptiko liniją buvo dešinės pusės sensorius 5](#_Toc57663107)

[3.1.6. Robotas linijos sekimo metu priekyje aptinka kliūtį naudojant ultragarso sensorių. 5](#_Toc57663108)

[3.2. Roboto arba robotų apvažiavimo scenarijus 6](#_Toc57663109)

[3.2.1. Robotas pradeda apvažiavimo manevrą 6](#_Toc57663110)

[3.2.2. Robotas apvažiavimo manevrą vykdo toliau 6](#_Toc57663111)

[3.2.3. Robotas neaptinka lėtesnio roboto su sensoriais 6](#_Toc57663112)

[3.2.4. Robotas aptinka apvažiojamą robotą naudojant priekinį ultragarso sensorių 6](#_Toc57663113)

[3.2.5. Robotas aptinka lėtesnį robotą naudojant kairį ultragarso sensorių. 6](#_Toc57663114)

[3.2.6. Robotas suka į kairę pusę 6](#_Toc57663115)

[4. Roboto aprašymas 7](#_Toc57663116)

[5. Roboto valdymo algoritmas 10](#_Toc57663117)

[6. Roboto algoritmo aprašymas 13](#_Toc57663118)

[7. Roboto valdymo algoritmo specifikacija Lua programavimo kalba 14](#_Toc57663119)

[8. Roboto modeliavimo(tyrimo) rezultatai 16](#_Toc57663120)

[9. Roboto valdymo programa 17](#_Toc57663121)

[10. Roboto valdymo eksperimentinis tyrimas 17](#_Toc57663122)

[11. Išvados (kiekvienas grupes narys formuoja savo išvadas ir nurodo savo indelį) 18](#_Toc57663123)

[Eligijus Kiudys 18](#_Toc57663124)

[Martynas Gidžiūna 18](#_Toc57663125)

[Simonas Brasas 18](#_Toc57663126)

[12. Naudotos literatūros sąrašas 19](#_Toc57663127)

Paveikslų sąrašas

[Pav. 1 Pridėtas papildomas kubas robotų atskyrimu 7](#_Toc57663128)

[Pav. 2 Pridėtas priekio ultragarso sensorius 7](#_Toc57663129)

[Pav. 3 Pridėtas kairiojo šono ultragarso sensorius 8](#_Toc57663130)

[Pav. 4 Robotai kuriuos apvažiuoja 9](#_Toc57663131)

[Pav. 6 Roboto algoritmas kintamųjų inicializavimas 10](#_Toc57663132)

[Pav. 7 Roboto apvažiavimo algoritmo dalis 11](#_Toc57663133)

[Pav. 8 Roboto apvažiavimo algoritmo paskutinė dalis 11](#_Toc57663134)

[Pav. 9 Roboto apvažiavimo algoritmo dalis, linijos sekimas 12](#_Toc57663135)

[Pav. 10 Algoritmo iliustracija 13](#_Toc57663136)

[Pav. 11 Reikšmių atnaujinimas realiu simuliacijos metu 14](#_Toc57663137)

[Pav. 12 Linijos sekimas 14](#_Toc57663138)

[Pav. 13 Aplenkimo algoritmo dalis CoppeliaSim aplinkoje 14](#_Toc57663139)

[Pav. 14 Aplenkimo algoritmo dalis CoppeliaSim aplinkoje 15](#_Toc57663140)

[Pav. 15 Aplenkimo algoritmo dalis CoppeliaSim aplinkoje 15](#_Toc57663141)

[Pav. 16 Algoritmo kintamųjų inicializavimas CoppeliaSim aplinkoje 15](#_Toc57663142)

[Pav. 17 Problema, delta laikas lygus 100ms 16](#_Toc57663143)

Lentelių sąrašas

[Lentelė. 1 Priekinio ultragarso sensoriaus naudojimas 5](#_Toc57663144)

[Lentelė. 2 Kairio ultragarso sensoriaus naudojimas 6](#_Toc57663145)

# Projektinio darbo užduotis

Robotų lenktynių sistema atliekama ,,CoppeliaSim‘‘ imituojamoje aplinkoje, naudojanti linijos sekimą ir ultra garso sensorius. Robotas turi nuolatos sekti juodą liniją, tačiau jeigu priekyje važiuojantis robotas važiuoja lėčiau galinis robotas turi pasisukti ir padidinti savo greitį, kad aplenktu pirmąjį robotą. Trasa bus vientisa ir turės posūkių.

# Užduoties ir reikalavimų analizė

Užduoties įvykdymui reikės:

* Pasirinkti vieną iš žinomų algoritmų, kuris būtu panašas į užduotį.
* Parašyti žinomą algoritmą ir jį modifikuoti daromai užduočiai.
* Modifikuoti robotą pagal algoritmo reikalavimus
* Pasidaryti trasą kuria robotas galėtu važiuoti ir pridėti lėtesnių robotų į trasą.

Robotas turės važiuoti sukurta trasa naudojant tris spalvos sekimo sensorius. Aptikus lėtesnį robotą autonomiškai jį apvažiuoti. Robotas turėtų atlikti paminėtus veiksmus neužkliaunant už lėtesnio roboto ir turėtu grįžti apvažiavęs lėtesnį robotą atgal į trasą.

# Roboto valdymo scenarijus

## Linijos sekimo scenarijus

### Dešinės pusės sensorius nebemato linijos

Robotas pasisuka į kairę pusę, kad robotas neišvažiuotu iš trasos

### Kairės pusės sensorius nebemato linijos

Robotas pasisuka į dešinę pusę, ištiesinti robotui.

### Vidurinis sensorius mato liniją

Robotas važiuoja tiesiai kol tik vidurinis sensorius aptinka linija

### Visi sensoriai linijos nebeaptinka ir paskutinis sensorius kuris aptiko liniją buvo kairės pusės sensorius

Robotas išvažiavęs iš trasos nepradėjus lenkimo manevro pradeda sukti į sensoriaus pusę kuris aptiko paskutinis trasą, kairę pusę.

### Visi sensoriai linijos nebeaptinka ir paskutinis sensorius kuris aptiko liniją buvo dešinės pusės sensorius

Robotas išvažiavęs iš trasos nepradėjus lenkimo manevro pradeda sukti į sensoriaus pusę kuris aptiko paskutinis trasą, dešinę pusę.

### Robotas linijos sekimo metu priekyje aptinka kliūtį naudojant ultragarso sensorių.

Lentelė. 1 Priekinio ultragarso sensoriaus naudojimas

|  |  |
| --- | --- |
| Atstumas iki kliūties | Veiksmas |
| 30< | Nustatomas apvažiavimo scenarijus |
| >=30 | Važiuoja pirmyn |

## Roboto arba robotų apvažiavimo scenarijus

### Robotas pradeda apvažiavimo manevrą

Robotas pasisuka į dešinę pusę ir toliau vykdo manevrą.

### Robotas apvažiavimo manevrą vykdo toliau

Robotas pavažiuoja pasirinktą laiko tarpą į priekį ir pasisuka per kitą laiko tarpą į kairę pusę.

### Robotas neaptinka lėtesnio roboto su sensoriais

Robotas pradeda sukti į kairę pusę xxx laiko pasisukęs pavažiuoja yyy laiko į priekį, šita būsena vyksta kol priekio arba kairio ultragarso sensorius aptinka lėtesnį robotą.

### Robotas aptinka apvažiojamą robotą naudojant priekinį ultragarso sensorių

Robotas pasisuka į dešinę pusę, kad neatsitrenktų į lėtesnį robotą.

### Robotas aptinka lėtesnį robotą naudojant kairį ultragarso sensorių.

Lentelė. 2 Kairio ultragarso sensoriaus naudojimas

|  |  |
| --- | --- |
| Atstumas iki kliūties | Veiksmas |
| 20< | Robotas pasisuka į dešinę pusę |
| 22< ir >19 | Robotas pasisuka lėčiau į dešinę pusę nei viršutinė būsena |
| >30 | Robotas pasisuka į kairę pusė |
| Nei vienas atstumas iš paminėtų neatitinka | Robotas važiuoja į priekį |

### Robotas suka į kairę pusę

Robotas apvažiavęs kitą robotą suka į kairę pusę kol pamato liniją. Roboto scenarijus yra pakeičiamas į linijos sekimo scenarijų kai linijos sekimo sensoriai pamato liniją.

# Roboto aprašymas

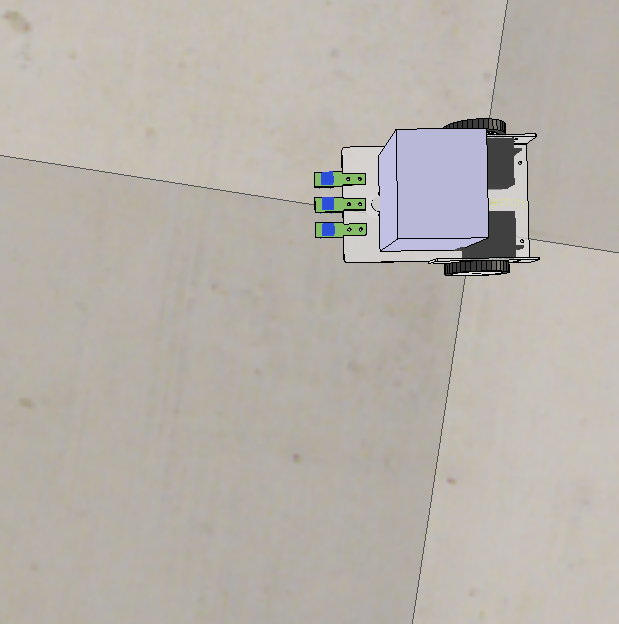
Projektiniam darbui yra naudojami trys robotai. Pirmasis robotas yra skirtas apvažiuoti kitus du robotus. Roboto bazei yra naudojamas duotas robotas line\_follower kuris yra Moodle sistemoje. Šiam robotui yra pridėti du papildomi ultragarso sensoriai ir modifikuotas kodas.

Roboto dalių paaiškinimas.

* „Line follower“ sekimo modulis, kuris atpažįsta spalvas.
* „Ultrasonic sensor“ sekimo modulis, kuris matuoja atstumą iki aptiktos kliūties.
* Servo tipo motorai, kurios galima kontroliuoti.

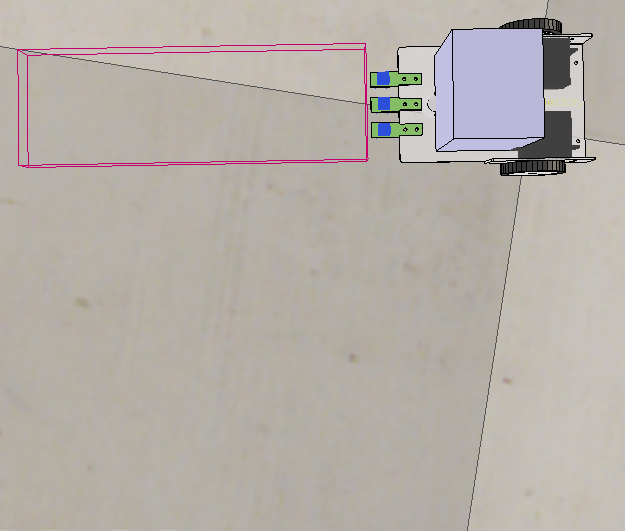
Duotas robotas Moodle sistemoje susidaro iš tokių dalių.

* Du servo tipo motorai.
* Trys spalvos matomumo sensoriai.
* Pagrindinis modelis.



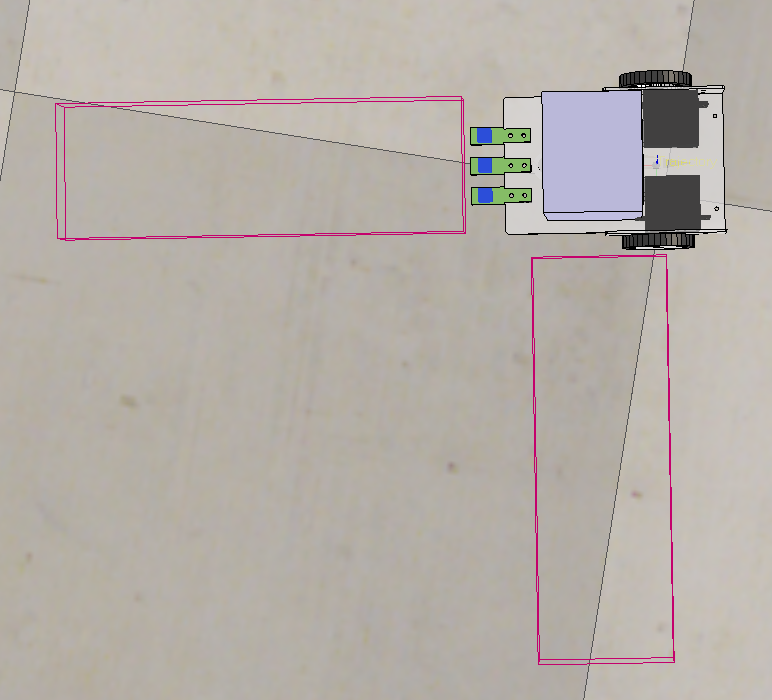
Pav. 1 Pridėtas papildomas kubas robotų atskyrimu

Pridėjome prie roboto papildomą kubą kuris padeda atskirti, kad robotas gali apvažiuoti priekyje esantį robotą.



Pav. 2 Pridėtas priekio ultragarso sensorius

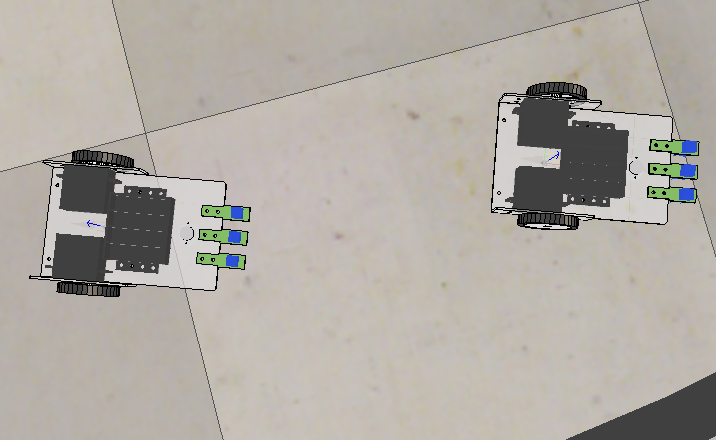
Roboto priekyje pridėjome ultragarso sensorių. Sensorius padeda robotui aptikti ar priekyje yra robotas kurį reikia apvažiuoti.



Pav. 3 Pridėtas kairiojo šono ultragarso sensorius

Robotui pridėjome antrąjį ultragarso sensorių kuris yra kairėje pusėje. Sensorius padeda aptikti, kada lėtesnis robotas yra kairiame šone. Robotas gali pasirinkti veiksmus pagal kairio sensoriaus atstumą iki lėtesnio roboto. Matome, kad robotas turi ratus kurie padeda jam judėti į priekį, bei priekyje tris linijos sekimo sensorius. Priekiniai sensoriai robotui padeda nustatyti į kurią pusę reikia sukti sekant trasos liniją.

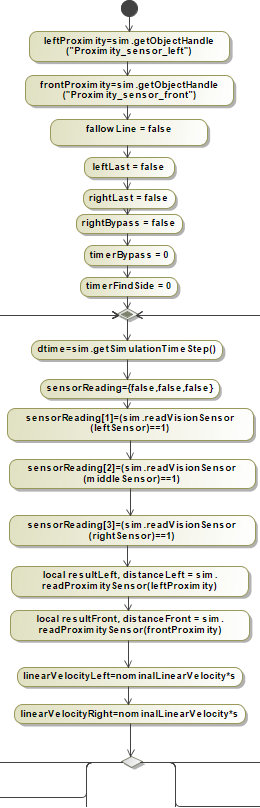
Kiti du robotai yra line\_follower duoto roboto kopijos kurių kodas yra modifikuotas.



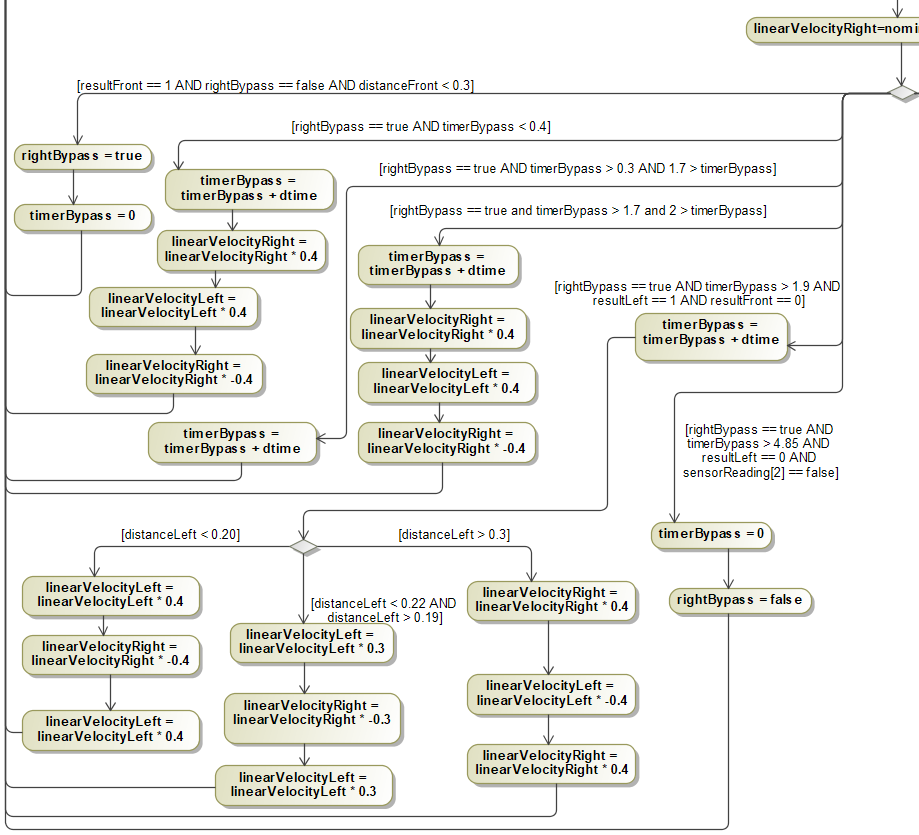
Pav. 4 Robotai kuriuos apvažiuoja

Naudojami linijos sekimo robotai kuriuos apvažiuoja kitas robotas. Šitie robotai yra nemodifikuoti, kadangi robotai turi sekti tik trasos liniją. Naudojamų linijos sekimo robotų kodas yra modifikuotas pagerinti linijos sekimo algoritmą.

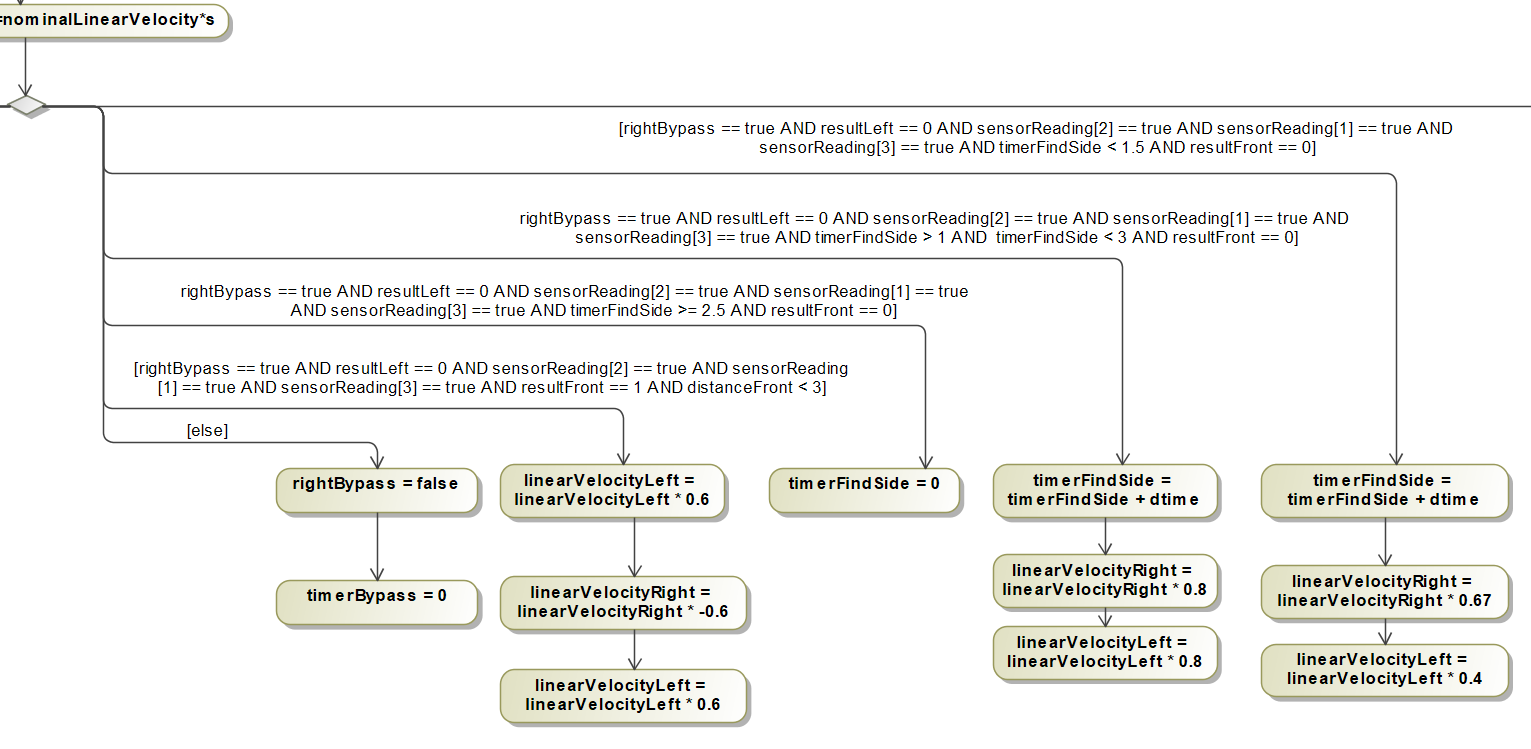
# Roboto valdymo algoritmas



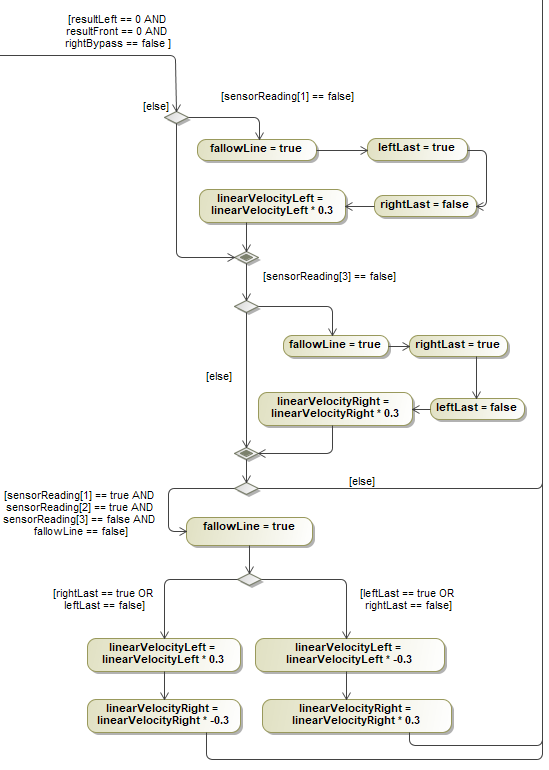
Pav. 6 Roboto algoritmas kintamųjų inicializavimas



Pav. 7 Roboto apvažiavimo algoritmo dalis



Pav. 8 Roboto apvažiavimo algoritmo paskutinė dalis



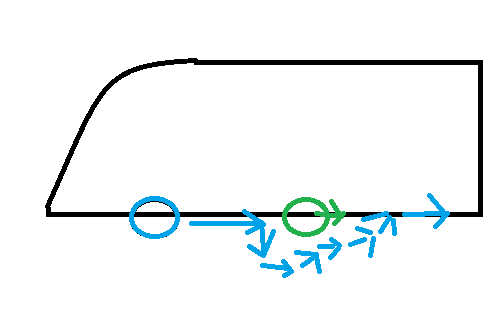
Pav. 9 Roboto apvažiavimo algoritmo dalis, linijos sekimas

Linijos sekimo algoritmas nesiskiria nuo modifikuoto algoritmo kuris yra naudojamas kituose dviejuose robotuose. Visų robotų ,,if [else] ir [sensorReading[1] == false]’’ dalių algoritmas yra vienodas.

# Roboto algoritmo aprašymas

Pasirinktas algoritmas yra vienas iš paprastesnių jei jis yra naudojamas ne dinamiškiems objektams. Dažniausiai šis algoritmas yra vadinamas „sienos sekėju“ ar kairės rankos taisyklė. Algoritmas pasižymi paprastumu, tačiau algoritmas veikia tik kai sienos yra sujungtos. Algoritmą reikėjo padaryti dinamiškai kadangi reikia apvažiuoti važiuojantį robotą.

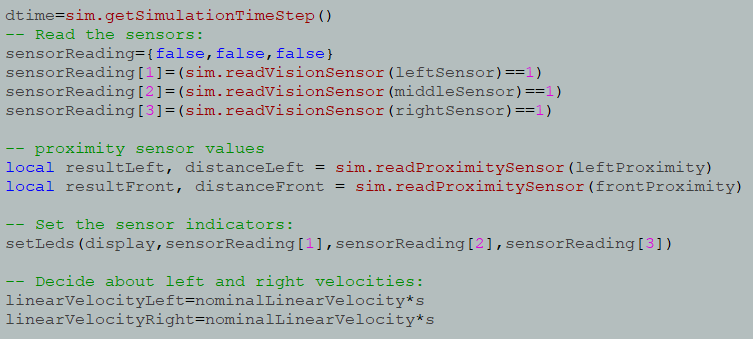
Sprendžiant šiuo būdu pradedant roboto apvažiavimą visada galime sekti kairės pusės ir priekio ultragarso sensorius, kad žinotume kada ir į kurią pusę robotas turi pasisukti. Šitas algoritmas veiks tik iš vienos pusės kadangi yra sekama tik viena pusė. Algoritmas garantuoja lėtesnio roboto apvažiavimą.



Pav. 10 Algoritmo iliustracija

Pasirinktas algoritmas yra dinamiškas. Algoritmas yra skirtas apvažiuoti kliūtis kurios važiuoja lėčiau arba išvengti priešinga puse važiuojančias kliūtis. Algoritmas nepasižymi paprastumu kadangi jis yra dinamiškas, greitas robotas privažiavęs lėtesnį robotą pasisuką į kairę pusę. Pasuktas robotas pavažiuoja į priekį ir pradeda ieškoti lėtesnio roboto su ultragarso sensoriais, važiuojant mažu lanku į kairę pusę. Sensoriams aptikus lėtesnį robotą, roboto ratų greitis yra pakeičiamas pagal atstumą nuo lėtesnio roboto. Roboto greitis pasikeičia taip, kad prisitaikytu prie lėtesnio roboto ir sugebėtų jį apvažiuoti. Pravažiavus lėtesnį robotą, tuomet greitesnis robotas pradeda važiuoti lanku į kairę pusę kol pamato liniją kurią turi sekti.

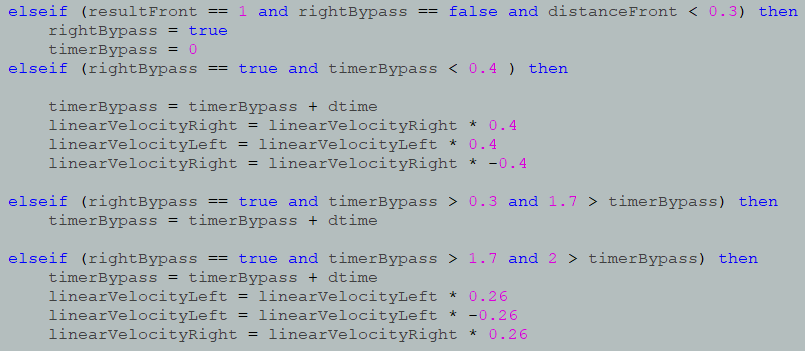
# Roboto valdymo algoritmo specifikacija Lua programavimo kalba



Pav. 11 Reikšmių atnaujinimas realiu simuliacijos metu



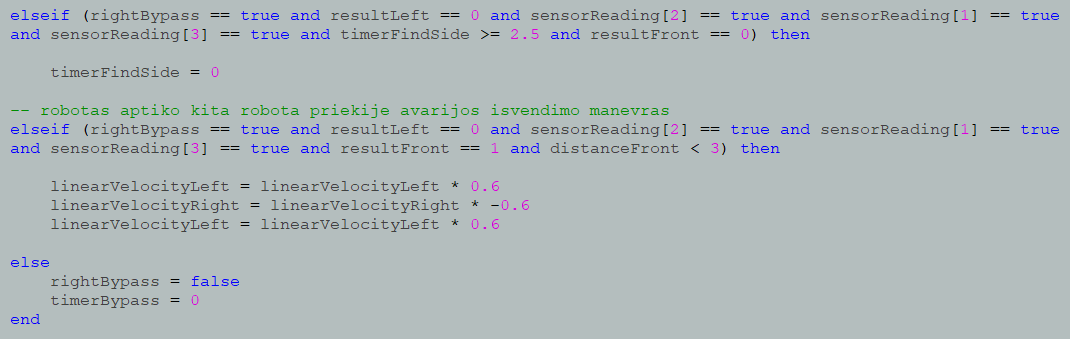
Pav. 12 Linijos sekimas



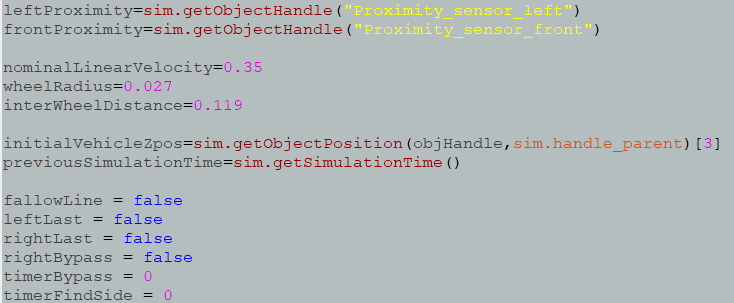
Pav. 13 Aplenkimo algoritmo dalis CoppeliaSim aplinkoje



Pav. 14 Aplenkimo algoritmo dalis CoppeliaSim aplinkoje



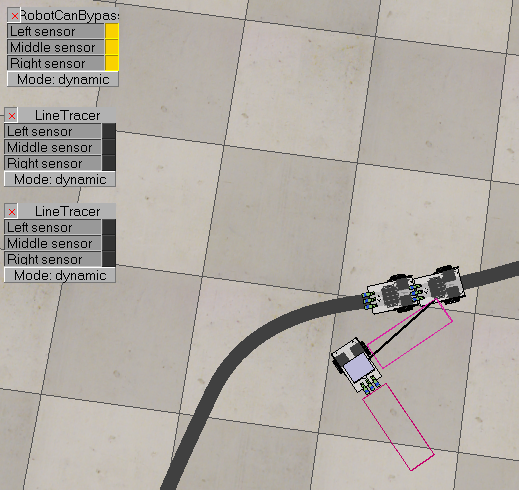
Pav. 15 Aplenkimo algoritmo dalis CoppeliaSim aplinkoje



Pav. 16 Algoritmo kintamųjų inicializavimas CoppeliaSim aplinkoje

# Roboto modeliavimo(tyrimo) rezultatai

Modeliavimas pavyko dalinai. Kadangi laiko skaičiavimas priklauso nuo delta laiko, pasitaiko situacijų, kuomet robotas apvažiuoja kliūtį, bet negrįžta į trasą kadangi per daug pasisuka į kairę pusę, sensoriams pamačius trasą robotas nebespėja į ją įvažiuoti ir ją pravažiuoja ( paveikslėlyje matome roboto trasos pravažiavimą).



Pav. 17 Problema, delta laikas lygus 100ms

Minėta situacija pasitaiko tik tada kai yra didelis delta laikas. Pamažinus delta laiką robotas veikia be problemų, kadangi robotas nepasisuka per daug.

# Roboto valdymo programa

Robotas nėra valdomas mygtuko paspaudimu. Pasirinkta užduotis yra autonomiškas sekimas linija ir apvažiavimas apie kitą judantį robotą. Roboto algoritmas sukurtas taip, kad veiktų visiškai savarankiškai, algoritmas įgyvendintas jog robotas būtu, besiadaptuojantis prie aplinkos.

# Roboto valdymo eksperimentinis tyrimas

Trasos aptikimui yra naudojami trys spalvos sensoriai. Sensoriai padeda robotui neišvažiuoti iš trasos kol robotas nepradeda apvažiavimo manevro. Robotui aptikus su kairiu sensoriumi kitokią spalvą nei juodą, robotas pasisuka į dešinę pusę. Robotui aptikus dešiniu spalvos sensoriumi ne juodą spalvą, robotas pasisuką į kairę pusę.

Roboto aplenkimui yra naudojami trys spalvos sekimo ir du ultragarso sensoriai. Greitas robotas privažiavęs lėtesnį robotą pasisuką į kairę pusę. Pasuktas robotas pavažiuoja į priekį ir pradeda ieškoti lėtesnio roboto su ultragarso sensoriais važiuojant mažu lanku į kairę pusę. Robotui su sensoriais aptikus lėtesnį robotą kurį reikia apvažiuoti yra keičiamas ratų greitis kuris priklauso nuo atstumo tarp jų dviejų. Roboto ratų greitis pasikeičia taip, kad robotas prisitaikytu prie lėtesnio roboto ir jį spėtų pravažiuoti. Pravažiavus lėtesnį robotą, robotas kuris naudoja apvažiavimo algoritmą pradeda važiuoti lanku į kairę pusę kol pamato liniją kurią turi sekti.

# Išvados (kiekvienas grupes narys formuoja savo išvadas ir nurodo savo indelį)

## Eligijus Kiudys

Iš esmės savo pasirinktą uždavinį pavyko atlikti. Robotas savarankiškai geba sekti trasą ir privažiavus lėtesnį robotą jį apvažiuoti. Tačiau pakeitus greitį dažnai robotas negrįžta į trasą. Sukurti programą, kurios pagalba, robotas naudodamas 2 ultragarso sensorius tvarkingai apvažiuotu lėtesnį robotą ir grįžtu į labirintą, gana sudėtinga dėl techninių galimybių ir pačio algoritmo sunkumo.

Prie projekto visi komandos nariai prisidėjo apylygiai. Visiems teko prisiliesti prie daugumos projekto atlikimo etapų: idėjos, roboto modifikavimo, programos kūrimo, trasos kūrimo, ataskaitos, bei prezentacijos.

## Martynas Gidžiūna

Roboto kūrime iškilo problemų, tame kad paprastuose duotose robotuose trūko komponentų, kaip vaizdo sensorių, dėl to reikėjo patiems sukurti savo ir pritaikyti projektui. Taipogi algoritmo rašymo etape, teko nuolatos iš naujo testuoti robotą, kadangi bet kokie maži kintamųjų pokyčiai galėjo priversti robotą nuvažiuoti nuo trasos arba nebegrįžti manevravimo atveju.

Prie projekto komandos nariai prisidėjo panašiai, kadangi visiems teko galvoti temą, mąstyti kaip rašyti algoritmą ir kartu programavo. Taipogi ataskaitą ir pristatymą visi pasidalinom darymo laikotarpiu apylygiai.

## Simonas Brasas

Kuriant robotą kuris gali aplenkti lėtesnį robotą užduotį atlikti pavyko. Roboto algoritmas įgyvendintas pakankamai korektiškai, privažiavęs mūsu atveju prie dviejų kitų robotų, greitesnis pasisuka ir apvažiuoja kliūtis, po to grįžta į kelią ir toliau važiuoja, posūkiai taipogi nesukuria problemų. Sunkiausia ir ilgiausiai užtruko algoritmo logikos kūrimas.

Visi prie projekto prisidėjo vienodai. Kiekvienas grupės narys ir galvojo kokia projekto užduotį reikia daryti, prisidėjo prie kodo rašymo ir mąstymo, roboto modelio kūrimo, ataskaitos ir pristatymo.

# Naudotos literatūros sąrašas

1. CoppeliaSim funkcijų sąrašas: <https://www.coppeliarobotics.com/helpFiles/en/apiFunctions.htm>
2. CoppeliaSim forumas: <https://forum.coppeliarobotics.com/>