

Demo Documentatie

HAN Arnhem

Versie 1

WOR-World

René van Eendenburg 561378

Derk Wiegerinck 567665

9 juni 2020

Inhoudsopgave

1	Inleiding	2
2	Benodigdheden	2
3	Bouwen van de broncode	2
4	Uitvoeren simulatie	2
5	Behaalde Eisen	3

1 Inleiding

Dit document beschrijft wat er nodig is om een simulatie te kunnen bouwen, het starten van de simulatie en het uitvoeren van een demo.

2 Benodigheden

Uw Linux distributie zal Robot Operating System (Ros) moeten ondersteunen. Kijk onder deze link of uw distributie wordt ondersteund.

Verder zijn de volgende onderdelen benodigd om de simulatie te bouwen en uit te voeren

- Ros Melodic
- GCC 7.4.0 of hoger

3 Bouwen van de broncode

Allereerst zult u een workspace moeten hebben. Dit kunt u doen door de volgende commando's uit te voeren

```
$ mkdir -p ~/catkin_ws/src
$ cd ~/catkin_ws/
$ catkin_make
$ source /opt/ros/<distro>/setup.bash
```

Daarna kunt u de packages in de zojuist aangemaakte src map kopiëren.

```
$ cp /directory/to/RobotSimulation/src/* ~/catkin_ws/src
$ catkin_make
$ source devel/setup.bash
```

4 Uitvoeren simulatie

Na het bouwen van de broncode, kan de simulatie worden gestart. Hiervoor opent u een terminal in de workspace en voert u de volgende stappen uit.

```
$ cd ~/catkin_ws
$ source devel/setup.bash
$ roslaunch al5d_simulation al5d_simulation.launch
```

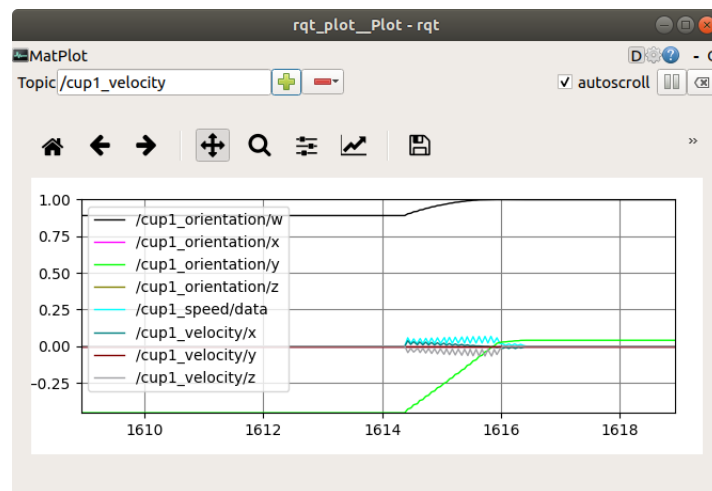
In deze demo is te zien hoe de beker wordt opgepakt. Daarna wordt de beker naar een nieuwe locatie gebracht en neergezet. Daarna zal de beker weer opgepakt worden en de robot de beker weer meenemen. Om te laten zien dat zwaartekracht ook werkt, wordt de beker losgelaten. Dit zorgt er wel voor dat de demo maar eenmalig kan worden uitgevoerd. Mocht deze nog een keer afgespeeld worden, moet de roslaunch en het demo script opnieuw worden gestart.

Daarnaast is het mogelijk om de bekerpositie aan te passen in de launch file. Zie hiervoor de Cup1Pos argument.

5 Behaalde Eisen

Criteria	Aangetoond door:
PA01-04	Alle code is opgebouwd volgens de ROS structuur en style guide. Verder zijn alle onderdelen OO opgebouwd
VS01-03	SSC32U berichten worden gestuurd naar het request topic en afgehandeld. Dit is zichtbaar in Rviz dankzij de bewegende robot arm.
VS04	De vertraging commando's per servo of globaal met tijd worden gebruikt in de Demo en worden realistisch afgehandeld.
VC01	De beker kan op een willekeurige positie worden gezet in de launchfile
VC03	Beker is zichtbaar gemaakt d.m.v. een marker
VC04-05	De relevante punten van de gripper worden aangeduid met een rode marker
VC06-07	De beker marker verandert van kleur wanneer opgepakt.
VC08	Zie demo
VC09-12	Alle relevante beker informatie is uit te lezen met rqt_plot. Lees met rqt_plot het cup1 topic uit of zie figuur 1
VC13	Zie demo
DI01	Alle relevante onderdelen worden opgestart met de launchfile
DI02	De bekerpositie is configurabel in de launchfile
DI04	Demoscript stuurt verschillende commando's met delays naar de demo

Tabel 1: Eisen



Figuur 1: rqt_plot van een gesimuleerde beker