Laboratorium z Anatomii Robotów

Dokumentacja projektu 1.

Maciej Święch

Piotr Szmurło

21.03.2021

***Cel projektu***

Celem projektu było stworzenie programu do sterowania symulatorem żółwia turtlesim. Sterowanie powinno być możliwe przy pomocy klawiszy zdefiniowanych przez użytkownika jako parametry węzła sterującego.

***Struktura projektu***

Struktura projektu składa się z dwóch węzłów – węzła symulatora żółwia (turtlesim\_node) oraz węzła wysyłającego sterowanie dla żółwia (turtle\_control\_talker).

Węzły komunikują się na temacie /turtle1/cm\_vel. Węzeł sterujący wysyła wiadomości typy geometry\_msgs/Twist, które węzeł żółwia interpretuje jako zmianę wartości zadanej prędkości. Wiadomości na temacie są publikowane co 0.01 sekundy dzięki czemu zmiana sterowanie powoduje zmianę ruchu żółwia praktycznie bez opóźnień. Przy mniejszej częstotliwości widać było opóźnienia co skutkowało tym, że zmiana sterowania wpływała na ruch żółwia dopiero po pewnym czasie.

W węźle turtle\_control\_talker zadeklarowano 4 parametry, którym domyślnie odpowiadają kolejno:

1. ‘up\_key’ – ‘w’
2. ‘down\_key’ – ‘s’
3. ‘left\_key’ – ‘a’
4. ‘right\_key’ – ‘d’

Parametry odpowiadają klawiszom, przy pomocy których można sterować ruchem żółwia. Naciśnięcie przycisku skutkuje:

1. ‘w’ – nadaniem żółwiowi prędkości liniowej o wartości 1,5 (jazda do przodu)
2. ‘s’ – nadaniem żółwiowi prędkości liniowej o wartości -1,5 (jazda do tyłu)
3. ‘a’ – nadaniem żółwiowi prędkości kątowej o wartości 1,5 (skręt w lewo)
4. ‘d’ – nadaniem żółwiowi prędkości kątowej o wartości -1,5 (skręt w prawo)

Do wczytywania klawiszy z klawiatury wykorzystaliśmy bibliotekę curses.

Program można wyłączyć klikając ‘q’. Wówczas zamykany jest program i czyszczony terminal.

Strukturę sterowania można włączyć albo przy pomocy komendy ros2 run (włączamy oddzielnie 2 węzły), lub przy pomocy utworzonego pliku turtle\_control\_launch.py. Wywołanie tego pliku skutkuje włączeniem obu węzłów.

turtle1/cmd\_vel

turtle\_control\_talker

turtlesim\_node