

Technical Report Ujian Akhir Semester Robotika

Disusun untuk memenuhi tugas Mata Kuliah Robotika dan Sistem Cerdas



Disusun oleh:

Muhammad Faisal Ramadhan (1103203227)

PROGRAM SARJANA TEKNIK KOMPUTER

FAKULTAS TEKNIK ELEKTRO

TELKOM UNIVERSITY

2022/2023

A. Pendahuluan

ROS2 (Robot Operating System 2) adalah sebuah framework perangkat lunak yang dirancang khusus untuk pengembangan dan operasi sistem robot. ROS2 menyediakan berbagai alat dan library yang memungkinkan pengguna untuk membangun sistem robot yang kompleks dengan mudah. Salah satu keunggulan utama ROS2 adalah kemampuannya dalam mendukung komunikasi antar modul dalam sistem robot secara terdistribusi, sehingga memungkinkan pengembang untuk mengintegrasikan berbagai komponen dan fungsionalitas dengan mudah.

Webots adalah sebuah simulator robotik yang sangat populer. Dalam lingkungan simulasi Webots, pengguna dapat membuat model 3D dari robot dan lingkungannya, serta menguji dan memvalidasi perilaku robot tersebut secara virtual. Webots menyediakan berbagai komponen dan lingkungan simulasi yang realistis, termasuk sensor, aktuator, dan fisika simulasi yang akurat. Dengan menggunakan Webots, pengembang dapat menguji dan mengembangkan algoritma kontrol robot sebelum melakukan implementasi fisik pada robot nyata.

Kendali dan simulasi robot merupakan dua aspek penting dalam pengembangan sistem robotik. Kendali robot melibatkan desain dan implementasi algoritma untuk mengendalikan pergerakan dan perilaku robot, termasuk navigasi, manipulasi objek, atau interaksi dengan lingkungan sekitar. Sementara itu, simulasi robot memungkinkan pengguna untuk menguji dan memvalidasi algoritma dan strategi kendali secara virtual sebelum diimplementasikan pada robot fisik. Simulasi memungkinkan pengembang untuk menghemat waktu dan biaya, serta mengidentifikasi dan memperbaiki masalah potensial sebelum melakukan eksperimen pada robot nyata.

Dalam pengembangan sistem robot, ROS2 dan Webots sering digunakan bersama-sama. ROS2 menyediakan kerangka kerja untuk mengintegrasikan komponen dan modul dalam sistem robot secara terdistribusi, sementara Webots menyediakan lingkungan simulasi yang realistis untuk menguji dan memvalidasi perilaku robot. Dengan menggunakan ROS2 dan Webots, pengembang dapat mengembangkan, menguji, dan memperbaiki algoritma kontrol robot sebelum mengimplementasikannya pada robot fisik, sehingga mengoptimalkan proses pengembangan dan meningkatkan kualitas sistem robot yang dihasilkan.

B. Penjelasan Skrip yang Digunakan

Kode terdiri dari sebuah skrip Python yang membuat sebuah node ROS2 dengan nama "my_node" dan mengimplementasikan beberapa fitur untuk

kendali dan simulasi robot. Kode tersebut menggunakan berbagai jenis pesan dan pustaka ROS2, sebagai berikut:

a. Impor Ketergantungan:

rcipy dan Node dari rcipy.node: Ini adalah pustaka inti ROS2 dalam Python yang diperlukan untuk membuat sebuah node ROS2.

Jenis pesan dari std_msgs.msg, geometry_msgs.msg, sensor_msgs.msg, moveit_msgs.msg, dan mavros_msgs.msg: Ini adalah jenis pesan yang digunakan untuk komunikasi antar node dan mengendalikan pergerakan robot.

b. Inisialisasi Node:

Dibuat kelas MyNode yang mewarisi dari Node, dan publisher, subscriber, dan variabel yang diperlukan diinisialisasi dalam konstruktor (`__init__`).

c. Fungsi Callback:

Fungsi callback didefinisikan untuk menangani pesan yang diterima pada topik "my_topic". Dalam kode ini, pesan yang diterima hanya dicetak ke konsol.

d. Fungsi Penerbitan (Publishing):

Fungsi `publish_twist` menerbitkan pesan Twist ke topik "cmd_vel" untuk mengendalikan kecepatan linear dan angular robot.

Fungsi `publish_range` menerbitkan pesan Range ke topik "distance" untuk memberikan pembacaan sensor jarak simulasi.

e. Fungsi Move Group dan Perencanaan Gerakan (Motion Planning):

Fungsi `send_move_group_goal` mengirimkan tujuan gerakan ke MoveIt Move Group untuk mengendalikan pergerakan robot.

Fungsi `send_motion_plan_request` mengirimkan permintaan perencanaan gerakan ke MoveIt Move Group untuk perencanaan lintasan.

f. Fungsi Kendali Drone:

Fungsi `send_position_target` menerbitkan pesan PositionTarget untuk mengendalikan posisi sebuah drone menggunakan mavros.

g. Fungsi Utama:

Fungsi `main` menginisialisasi konteks ROS2, membuat sebuah instance dari kelas MyNode, mengatur timer untuk menjalankan fungsi penerbitan dan kendali, menjalankan node, dan menutup konteks ROS2 saat selesai.

C. Implementasi Penggunaan Skrip

Untuk menggunakan kode yang diberikan, kita harus mengikuti langkah-langkah berikut:

1. Instal ROS2 dan dependensinya.
2. Buat workspace ROS2 dan bangun workspace tersebut.
3. Salin kode ke dalam sebuah berkas skrip Python (misalnya, drone.py).
4. Buka terminal, arahkan ke workspace ROS2, dan sumberkan berkas setup.
5. Jalankan skrip menggunakan perintah: `python3 drone.py`

D. Kesimpulan

Laporan teknis ini menjelaskan implementasi sebuah node ROS2 untuk kendali dan simulasi robot dengan menggabungkan contoh kode dari bab-bab yang berbeda dalam buku "Mastering ROS2". Kode yang diberikan menunjukkan berbagai fungsionalitas seperti komunikasi pesan, kendali pergerakan robot, perencanaan gerakan, dan kendali drone.