

ROS day1 hw1 보고서

2025407012/로봇학부/송연우

목차:

1. Topic

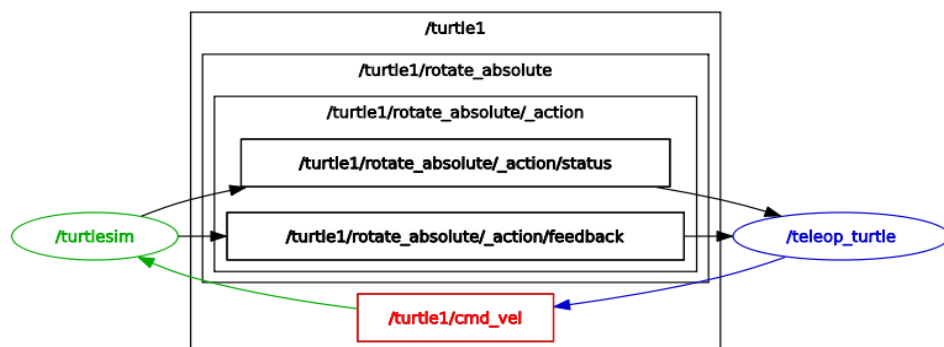
2. Turtlesim

1. Topic

토픽은 비동기식 단방향 메시지 송수신 방식으로, 메시지를 발행하는 Publisher와 메시지를 수신하는 Subscriber, 중간에서 메시지의 전달 통로가 되는 Topic으로 구성됩니다. ros2에서도 topic을 지원하며 로봇 통신에서 많이 활용됩니다. Publisher node와 Subscriber node는 같은 topic 이름을 사용하고, Publisher node에서 메시지를 발행하고 있어야 Subscriber node에서 받을 수 있습니다. 이때 Publisher node가 메시지를 발행해도 Subscriber node에서 무조건 받는 통신은 아니며, 받을지 말지를 선택할 수 있습니다.

2. Turtlesim

turtlesim은 다양한 통신을 사용하는 프로그램으로 사용자의 입력을 받아 화면에 뜨는 거북이를 움직이는 방식으로 작동합니다. turtlesim에서 주로 사용되는 topic은 /turtle1/cmd_vel로, 아래 그림과 같이



rqt_graph 명령어로 현재 실행 중인 node, topic, action을 gui 형식으로 확인할 수 있습니다. 이때 turtlesim에서의 거북이가 화면에서 움직이는 동작은 중간 토픽인 /turtle1/cmd_vel를 Subscribing하고 있는 turtlesim에 전달되어 반영됨을 알 수 있습니다.

turtlesim을 실행하기 위해서는 turtlesim_node와 turtle_teleop_key를 각각의 터미널에서 실행해야 하고, key를 실행한 터미널에서 방향키를 누르면 turtlesim에 움직이는 거북이의 모습이 나타납니다. key를 통해 명령을 입력할 수도 있지만, topic인 /turtle1/cmd_vel에 직접

메시지를 보내 turtlesim 화면에 변화를 줄 수도 있습니다. 토픽을 발행하는 명령어로 `ros2 topic pub --once /turtle1/cmd_vel geometry_msgs/msg/Twist "{linear: ~~~ angular: ~~~}"`를 입력하면 사용자가 입력한 대로 움직입니다. 이때 Twist 는 float64자료형을 가지는 메시지 타입으로 Linear 부분, angular 부분으로 나뉘어지며 각각 3축을 가지고, 로봇을 움직이는 등의 작업에서도 이용됩니다.

다음으로 turtlesim 을 터미널 토픽 발행을 통해 움직이겠습니다.

```
yu@yu:~$ ros2 topic pub --rate 1 /turtle1/cmd_vel geometry_msgs/msg/Twist "{linear: {x: 2.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 1.8}}"
```

publisher: beginning loop

publishing #1: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=2.0, y=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=1.8))

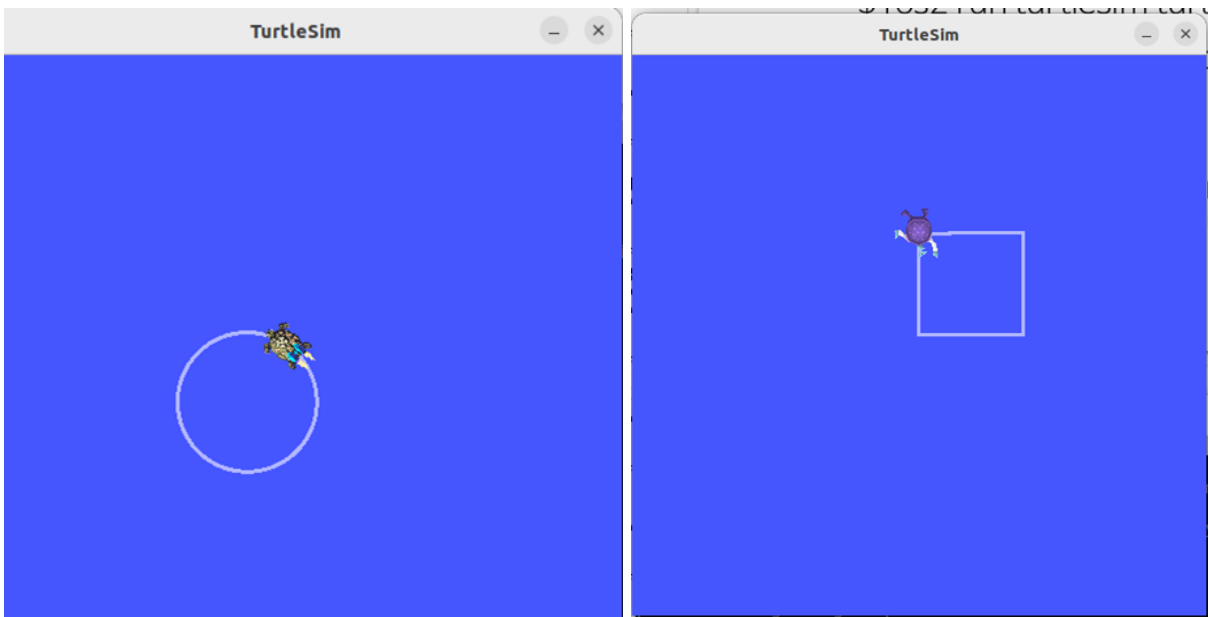
publishing #2: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=2.0, y=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=1.8))

publishing #3: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=2.0, y=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=1.8))

publishing #4: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=2.0, y=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=1.8))

^Cyu@yu:~\$

`ros2 topic pub --once /turtle1/cmd_vel geometry_msgs/msg/Twist "{linear: {x: 2.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 1.8}}"` 명령어를 입력하면 거북이가 원의 일부를 그리며 돌게 되는데, 3번 정도 반복하면 완전한 원을 그릴 수 있습니다. 이때 z 값이 약 1.57일 때 거북이가 90도 회전을 하게 되므로 사각형을 그리는 명령은 `"{linear: {x: 2.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 0.0}}"`과 `"{linear: {x: 0.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 1.56}}"`을 번갈아 입력하는 것과 같습니다.



다음으로 삼각형을 그리기 위해서는 이동하고 120도를 도는 과정을 반복하게 되므로 명령은 `"{linear: {x: 2.0, y: 0.0, z: 0.0}, angular: {x: 0.0,`

y: 0.0, z: 0.0}}"과 "{linear: {x: 0.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 2.08}}"을 번갈아
입력합니다.

