

Robot Guidance with Head Movements

Versiyon : v1.0
Tarih : 31/08/2020
Hazırlayan : Elif GENÇ

DÖKÜMAN REVİZYON SAYFASI

VERSIYON	TARİH	SAYFA	AÇIKLAMA
v1.0	31/08/2020	10	Ubuntu Bionic Beaver ve Ros Melodic Kurulumu, Ros Beginner Tutorials

İÇİNDEKİLER

1. TANITIM VE KAPSAM	4
2. KULLANILAN MATERYALLER	4
3. ÖNERİLEN YÖNTEM / PROBLEM ÇÖZÜM AŞAMALARI	4
4. GERÇEKLENEN GÖREVİN AŞAMALARI	5
5. SONUÇLAR	9
6. KAYNAKÇA	10

1. TANITIM VE KAPSAM

Bu çalışmalar kapsamında Ubuntu 18.04, ROS Melodic dağıtımının kurulumları ve konfigürasyonlarının nasıl gerçekleştirildiği ve sistem gereklilikleri araştırılmıştır. Sistem gereksinimleri temin edildiği web sitelerinden kontrol edilmiştir.[1][2] Daha sonra bu gereksinimlerin karşılandığı kontrol edilmiştir, daha sonra kurulum ve konfigürasyonlar gerçekleştirilmiştir. Daha sonra başlangıç seviyesinde bulunan ROS komutları ve terminolojileri öğrenilmiştir. Bu rapor içerisinde kullanılan materyaller, yaşanan problemler ve çözüm yöntemleri, görevlerin aşamaları detaylı bir şekilde anlatılmıştır.

2. KULLANILAN MATERYALLER

Çalışmalarımız boyunca kullanacağımız materyaller; Ubuntu 18.04.5, ROS Melodic Morenia Dağıtımıdır.

Ubuntu, Linux çekirdeği temel alınarak geliştirilen açık kaynak kodlu, özgür ve ücretsiz bir işletim sistemidir. Bu kapsamda kullanacağımız versiyon Ubuntu 18.04 Bionic Beaver' dır. Ubuntu 18.04 Bionic Beaver, Ubuntu' nun LTS sürümlerinden biri olan ve Gnome masaüstü ortamına sahip versiyondur. LTS (Long Term Support) olarak adlandırılan yani uzun süreli desteğe sahip olan sürümdür. [3]

ROS dağıtımı, sürümlenmiş bir ROS paketleri grubudur. Bunlar Linux dağıtımlarına (ör. Ubuntu) benzemektedir. ROS dağıtımlarının amacı, geliştiricilerin bir sonraki sürümü hazır olana kadar çalışmasını nispeten istikrarlı bir kod tabanı ile yapmalarına izin vermektir. [4] Bu çalışmada kullanacağımız ROS dağıtımı, ROS Melodic Morenia' dır. Melodic Morenia dağıtımı, öncelikle Ubuntu 18.04 (Bionic) sürümüne yöneliktir.

ROS kurulumu ve gerekli konfigürasyonlar yapılırken işletim sistemine yüklenen kütüphaneler ve programlar: ROS, rqt, rviz, robot - generic kütüphaneleri, 2D/3D simülörler, navigasyon ve 2D/3D perspektif.

3. ÖNERİLEN YÖNTEM / PROBLEM ÇÖZÜM AŞAMALARI

Yaşanılan iki tane problem vardır. Bunlardan biri ROS Melodic Morenia dağıtımının Ubuntu 18.04.2 üzerine kurulması sırasında alınan bozuk paket hatalarıdır. Bu bozuk paketler için alınan hatayı düzeltmek için araştırılmalar yapılmıştır. Bulunan çözüm yolları uygulanmıştır ve bu çözüm yolları bu hatanın giderilmesi konusunda yetersiz kalmıştır. Daha sonra yeni bir çözüm yolu araştırılmıştır ve Ubuntu 18.04.5 sürümüne yükseltilmesiyle hata giderilmiştir ve ROS Melodic Morenia dağıtımının kurulumu tamamlanmıştır.

Yaşanan ikinci problemde çalışma ortamının kurulması aşamasında oluşturulması için kullanılan `"$ mkdir -p ~/catkin_ws/src"` komut istenilen çalışma ortamını oluşturamamıştır. Bu komut uygulanmadan önce terminalde `"$ printenv | grep ROS"` komutu çalıştırılarak problem giderilmiştir.

4. GERÇEKLENEN GÖREVİN AŞAMALARI

Bu aşamada temel ROS komutları ve öğrenilen terminolojiler açıklanmıştır. [5][6]

- ROS komutlarının bilgisayarda aktif edilmesini sağlayan komut.

```
$ source /opt/ros/melodic/setup.bash
```

- ‘catkin’ çalışma alanının kurulumunu ve çalışma alanının içini catkinin kendi standart araçlarıyla doldurulmasını sağlar.

```
$ mkdir -p ~/catkin_ws/src  
$ cd ~/catkin_ws/  
$ catkin_make
```

- “roscpp” isimli paketin bulunduğu path ile ilgili bilgileri veren komut

```
$ rospack find roscpp
```

- “roscpp” isimli paketin path konumuna erişmek için kullanılan komut

```
$ roscd roscpp
```

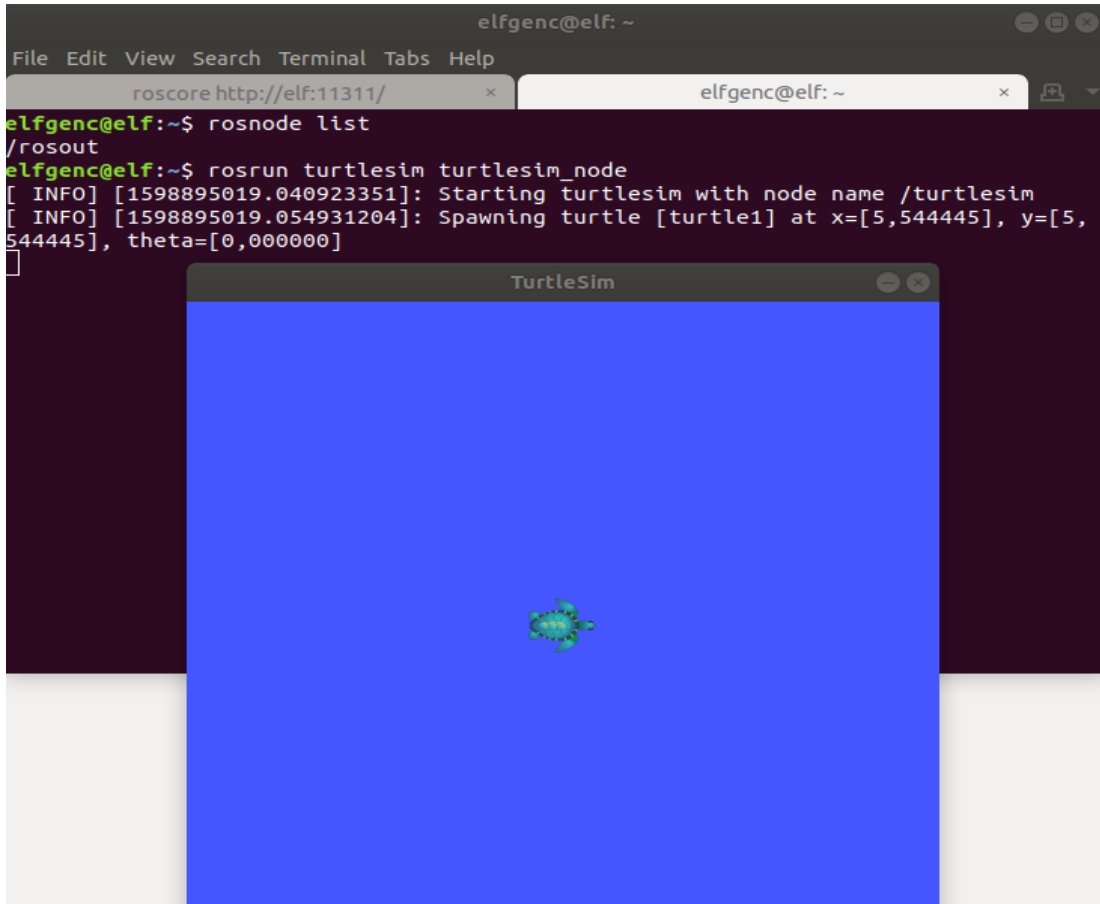
- catkin paketini ‘catkin_create_pkg’ komutu ile daha önce açtığımız catkin çalışma alanının içindeki ‘src’ klasörüne kurulması için çalıştırılması gereken kodlar bunlardır.

```
$ cd ~/catkin_ws/src  
$ catkin_create_pkg beginner_tutorials std_msgs rospy roscpp
```

- src klasörüne kurulan paketimizin catkin_make komutuyla yapılandırılması için gerekli komutlar bunlardır.

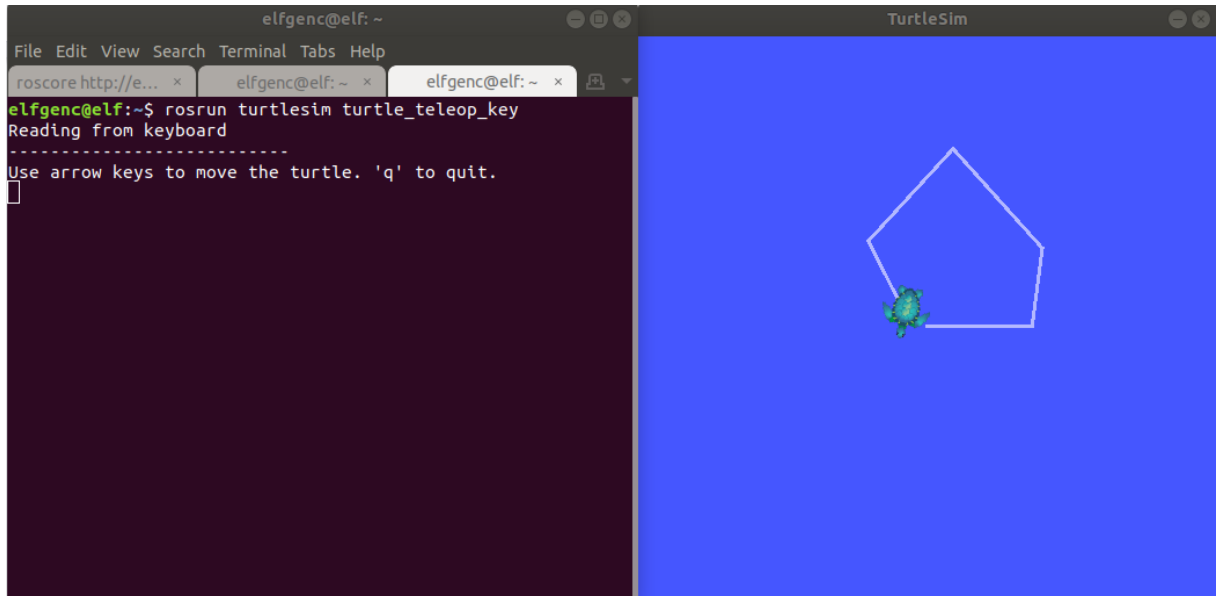
```
$ cd ~/catkin_ws  
$ catkin_make
```

- ROS içerisindeki temel yapılar: Düğümler(Nodes), Konular(Topics) ve Mesajlardır(Messages). Düğümler birbiriyle iletişim kurarken mesajları kullanırlar ama doğrudan mesajları birbirine gönderemezler. Mesajlar konular üzerinden düğümlere aktarılır. Örnek olarak kullanılan düğüm “turtlesim_node” isimli düğümdür. Bu düğümün çalıştırılması için öncelikle ROS’ un çalıştırılması gerekmektedir. Ros sisteminin çalıştırılması için “\$ roscore” komutunun çalıştırılması gerekmektedir. Daha sonra bu komutun çalıştırıldığı terminal kapatılmadan başka bir terminale geçerek “\$ rosrunc turtlesim turtlesim_node” komutunun çalıştırılması sonucu ekrana şekil 1’ deki görüntü gelmektedir.



Şekil 1: turtlesim_node

- Açılan ‘turtlesim’ düğümü ile ekrana gelen robotun klavye kontrolünü sağlamak için “*\$ roslaunch turtlesim turtle_teleop_key*” komutu başka bir terminalde çalıştırılmıştır. Şekil 2’ de klavye kullanarak robotun hareket ettiği path görülmektedir.



Şekil 2: turtle_teleop_key

- Klavye ile robota verilen yönler doğrultusunda robotta oluşan koordinat ve ivme değişkenlerinin düğümler arasındaki mesajlar olarak aktarılmasının görüntüsü şekil 3’ te gösterilmiştir. Bu mesajlara bakabilmek için terminalde çalıştırılması gereken komut “\$ rostopic echo /turtle1/pose” şeklindedir.

```

elfgenc@elf: ~
File Edit View Search Terminal Tabs Help
elfgenc... x roscore ... x elfgenc... x elfgenc... x elfgenc... x elfgenc... x
y: 5.88027477264
theta: 0.670444071293
linear_velocity: 0.0
angular_velocity: 0.0
---
x: 6.20600795746
y: 5.88027477264
theta: 0.670444071293
linear_velocity: 0.0
angular_velocity: 0.0
---
x: 6.20600795746
y: 5.88027477264
theta: 0.670444071293
linear_velocity: 0.0
angular_velocity: 0.0
---
x: 6.20600795746
y: 5.88027477264
theta: 0.670444071293
linear_velocity: 0.0
angular_velocity: 0.0
---

```

Şekil 3: Robot Konum Ve Hız Bilgileri

- Düğümler arasında gönderilen mesajların boyutu, süresi, mesaj sayısı, mesaj türleri gibi bilgileri kaydetmek için veri kaydı dosyası oluşturmak gerekmektedir. Bu veri kaydı dosyalarını içeren klasörü oluşturmak için “\$ mkdir ~/bagfiles” komutu kullanılmaktadır. Daha sonra bu klasör içerisine erişebilmek için “\$ cd ~/bagfiles” komutu çalıştırılır. Ardından veri kaydını başlatmak için “\$ rosbag record -a” komutu çalıştırılır. Kaydettiğimiz dosyanın bilgilerini kontrol etmek için “\$ rosbag info 2020-08-31-21-01-39.bag” komutu kullanıldı. Bu komut ile veri kaydı dosyasının bilgileri şekil 4’te gösterilmiştir.

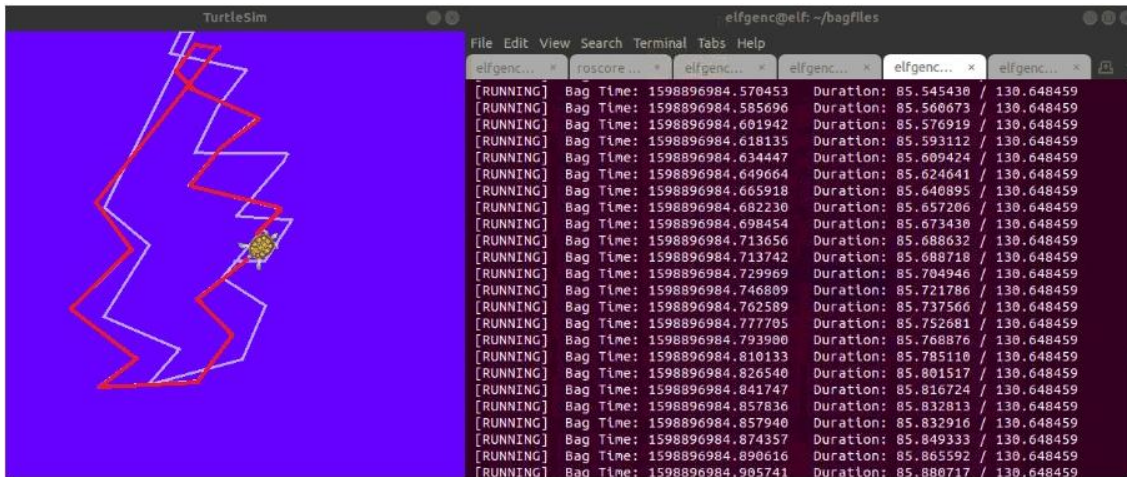
```

elfgenc@elf: ~/bagfiles
File Edit View Search Terminal Tabs Help
elfgenc... x roscore... x elfgenc... x elfgenc... x elfgenc... x elfgenc... x
[ INFO] [1598897072.420745642]: Subscribing to /turtle1/pose
^Celfgenc@elf:~$ rosbag info bagfiles
ERROR reading bagfiles: [Errno 21] Is a directory: 'bagfiles'
elfgenc@elf:~$ cd bagfiles/
elfgenc@elf:~/bagfiles$ rosbag info 2020-08-31-21-01-39.bag
path:          2020-08-31-21-01-39.bag
version:       2.0
duration:      2:10s (130s)
start:         Aug 31 2020 21:01:39.03 (1598896899.03)
end:           Aug 31 2020 21:03:49.67 (1598897029.67)
size:          1.1 MB
messages:      16709
compression:   none [2/2 chunks]
types:         geometry_msgs/Twist [9f195f881246fdafa2798d1d3eebca84a]
               roscpp_msgs/Log     [acffd30cd6b6de30f120938c17c593fb]
               turtlesim/Color      [353891e354491c51aabe32df673fb446]
               turtlesim/Pose       [863b248d5016ca62ea2e895ae5265cf9]
topics:        /rosout              25 msgs      : roscpp_msgs/Log      (2 connections)
               /rosout_agg          21 msgs      : roscpp_msgs/Log
               /turtle1/cmd_vel     366 msgs     : geometry_msgs/Twist
               /turtle1/color_sensor 8148 msgs    : turtlesim/Color
               /turtle1/pose        8149 msgs    : turtlesim/Pose
elfgenc@elf:~/bagfiles$

```

Şekil 4: Veri Kayıt Dosyası Bilgisi

- Kaydedilen verilerin robotun bulunduğu yerden tekrar başlatılarak hareket edilmesi için “\$ rosbag play 2020-08-31-21-01-39.bag” komutu kullanıldı. Tekrar oynatılan veri kayıt dosyası sonucunda robotumuz şekil 5’ teki kırmızı ile gösterilen yolu takip etmiştir.



Şekil 5: Kayıtlı Robot Verisinin Oynatılması

5. SONUÇLAR

Yapılan çalışmalar kapsamında ROS Melodic dağıtımının Ubuntu 18.04.5 LTS versiyonuna başarılı bir şekilde yüklenebilmesi sağlanmıştır. Ayrıca bu çalışmalar sırasında karşılaşılan en büyük problem ROS Melodic dağıtımının Ubuntu 18.04.2 versiyonunda kurulamaması ve bu aşamada python-roscdistro-module isimli paketteki bozulmalar sebebiyle alınan hatanın düzeltilmemesidir. Bu hatanın giderilmesi için gerekli araştırmalar yapılmıştır ve Ubuntu sürümünün değiştirilmesine karar verilmiştir. .Daha sonra ROS Melodic’ in temel bilgileri ve başlangıç seviyedeki ROS komutlarının öğrenilmesi gerçekleştirilmiştir. ROS komutlarını kullanarak bir klasör üzerinde ROS’un nasıl başlatılacağı ve bunun gereklilikleri öğrenilmiştir. Ros’ un temelini oluşturan düğümler(nodes), konular(topics) ve mesajlar(messages) öğrenilmiştir ve bunlarla ilgili örnekler yapılmıştır. Bu çalışmalar kapsamında turtle düğümlerini kullanarak klavye düğümü ile robot hareket ettirildi. Robotun hareketleri ile değişen konum bilgileri gözlemlendi ve bu bilgiler bir dosyaya kaydedildi. Daha sonra kaydedilen dosyadaki verileri kullanarak robotun izlediği yol tekrar oluşturuldu.

6. KAYNAKÇA

- [1] <https://help.ubuntu.com/community/Installation/SystemRequirements> [Erişim tarihi: Ağustos, 2020]
- [2] <http://wiki.ros.org/melodic> [Erişim tarihi: Ağustos, 2020]
- [3] https://wiki.ubuntu-tr.net/index.php?title=Ubuntu_nedir%3F [Erişim tarihi: Ağustos, 2020]
- [4] <http://yapbenzet.kocaeli.edu.tr/ros-dagitimlari/> [Erişim tarihi: Ağustos, 2020]
- [5] <http://wiki.ros.org/tr/ROS/Tutorials> [Erişim tarihi: Ağustos, 2020]
- [6] <http://wiki.ros.org/ROS/Tutorials> [Erişim tarihi: Ağustos, 2020]