



Gazebo ve ROS

ROS-GAZEBO Arası Etkileşim ve Örnek Uygulamalar
ROS Uygulamalı Eğitimleri 2019, Eskişehir



Supported by ROSIN - ROS-Industrial Quality-Assured Robot Software Components.
More information: rosin-project.eu



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 732287.

İçerik

- **Gazebo nedir?**
 - **Gazebo Bileşenleri**
 - Dünya Dosyaları
 - Model Dosyaları
 - Ortam Değişkenleri
 - Gazebo Sunucusu
 - Gazebo İstemcisi
 - Eklentiler
 - **Gazebo Uygulamaları**
 - Gazebo Kurulumu
 - Gazebo Çalıştırılması
 - Çalışma Ortamının Hazırlanması
 - Dünya Oluşturulması
 - Model Oluşturulması
 - Mesh Giydirme
 - Sensör Ekleme
 - **ROS ile Kontrol**
-

Gazebo nedir?

Robotların iç ve dış ortamlarda simülasyonu için gürbüz fizik motoru, yüksek kalitede grafikler ve grafik arayüzü ile gerekli altyapıyı sunar.

Gazebo;

- ODE, Bullet, Simbody ve DART gibi yüksek performansa sahip fizik motorlarının kullanımlarını,
- OGRE sayesinde ortamların gerçekçi bir şekilde oluşturulmasını,
- Sensör ve sensör verilerinin kullanımlarını,
- Mevcut robot modellerinin kullanımlarını veya kendi robot modelini oluşturma imkanını

sağlamaktadır.



Gazebo Bileşenleri

Gazebo bünyesinde gerçekleştirilen bir simülasyon aşağıdaki öğeleri içermektedir:

- Dünya dosyaları
- Model dosyaları
- Ortam Değişkenleri
- Gazebo sunucusu
- Gazebo istemcisi
- Eklentiler (Plugins)



Dünya Dosyaları

Dünya dosyaları, simülasyon ortamındaki;

- robot,
- sensör,
- ışık,
- nesne

gibi ortam elemanlarını tanımlamak için kullanılmaktadır.

Bu dosyalar SDF isimli tanımlama biçimi kullanılarak oluşturulmakta ve genel itibariyle .world uzantısına sahip olmaktadır.



Dünya Dosyaları

Çeşitli dünya dosyaları Gazebo kurulumuyla beraber gelmekte ve

```
<install_path>/share/gazebo-<version>/worlds
```

içerisinde yer almaktadır.

Aynı zamanda SDF tanımlama biçimi kullanılarak kullanıcı tanımlı dünya dosyaları oluşturmak da mümkündür.



Dünya Dosyaları

Örnek bir dünya dosyası olan "empty.world" aşağıdaki gibi bir içeriğe sahiptir.

```
<?xml version="1.0" ?>
<sdf version="1.5">
  <world name="default">
    <!-- A global light source -->
      <include>
        <uri>model://sun</uri>
      </include>
    <!-- A ground plane -->
      <include>
        <uri>model://ground_plane</uri>
      </include>
    </world>
  </sdf>
```



Model Dosyaları

Dünya dosyaları gibi SDF tanımlama biçimi kullanmaktadır ve sadece

```
<model>  
  ...  
</model>
```

şeklinde tag içermektedir.

Bu dosyaların kullanımındaki amaç, modellerin tekrar kullanılabilirliğini sağlamak ve dünya dosyalarını basitleştirmektir.



Model Dosyaları

Oluşturulan bir model, dünya dosyası içerisinde aşağıdaki gibi kullanılabilir.

```
<include>  
    <uri>model://model_file_name</uri>  
</include>
```

Simülasyon ortamı için, online veritabanında yer alan veya eski Gazebo versiyonlarında kurulumla beraber gelen model dosyaları kullanılabilir. Ayrıca, kullanıcı tarafından tanımlanan model dosyaları da kullanılabilir.



Ortam Değişkenleri

Gazebo, dosyaları konumlandırıp erişebilmek ve sunucu ile istemci arasındaki ilişkiyi kurabilmek adına çeşitli ortam değişkenleri kullanmaktadır. Kurulumla beraber ortam değişkenleri derlenmiş olarak gelir.

Varsayılan ortam değişkenleri aşağıdaki shell script (.sh dosyası) içerisinde yer almaktadır.

```
<install_path>/share/gazebo/setup.sh
```

Gazebo'nun dikkate alacağı ortam değişkenleri değiştirilmek istenirse ilgili shell script dosyası

```
source <install_path>/share/gazebo/setup.sh
```

ile kaynak gösterilerek, isteğe bağlı olarak modifiye edilmektedir.



GAZEBO

Ortam Değişkenleri

Örnek bir script dosyası içeriği aşağıda gösterilmektedir.

```
export GAZEBO_MASTER_URI=http://localhost:11345
export GAZEBO_MODEL_DATABASE_URI=http://gazebosim.org/models
export GAZEBO_RESOURCE_PATH=/usr/share/gazebo-
7:/usr/share/gazebo_models:${GAZEBO_RESOURCE_PATH}
export GAZEBO_PLUGIN_PATH=/usr/lib/x86_64-linux-gnu/gazebo-7/plugins:${
GAZEBO_PLUGIN_PATH}
export GAZEBO_MODEL_PATH=/usr/share/gazebo-7/models:${
GAZEBO_MODEL_PATH}
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/usr/lib/x86_64-linux-
gnu/gazebo-7/plugins
```



Gazebo Sunucusu

Gazebo sunucusu, Gazebo'nun tüm işlem yükünü üstlenen kısmıdır. Kendisine verilen bir dünya dosyasını ayrıştırır ve ardından bir fizik ve sensör motoru kullanarak dünyayı simüle eder.

Gazebo sunucusu herhangi bir grafik arayüzü içermemektedir.



Gazebo Sunucusu

Gazebo sunucusu terminalde

```
gzserver
```

komutuyla yalın olarak başlatılabileceği gibi

```
gzserver worlds/empty.world
```

komutunda olduğu gibi bir dünya dosyası (bu örnekte Gazebo ile birlikte gelen bir dünya dosyası) ile birlikte de başlatılabilmektedir.



Gazebo İstemcisi

Gazebo istemcisi, çalışan bir Gazebo sunucusuna bağlanarak ve ortama ait elemanların görselleştirilmesi görevini üstlenmektedir.

Aynı zamanda kullanıcıya çalışan bir simülasyon üzerinde değişiklik yapma fırsatı sunmaktadır.

Gazebo sunucusu, terminalde

`gzclient`

komutuyla yalın olarak başlatılabilmektedir.



Eklentiler (Plugins)

Eklentiler, ortak kütüphane dosyası olarak derlenen ve simülasyona eklenen kod parçaları olarak ele alınmaktadır. Eklentilerin, standart C++ sınıfları üzerinden tüm Gazebo işlevselliğine erişimleri mümkündür.

Eklentiler, Gazebo'nun her yönüyle kontrol edilmesine imkan tanımakta ve çalışan bir sisteme eklenip çıkarılabilmektedir.



Eklentiler (Plugins)

Programlı bir şekilde simülasyonda bir şeyler değiştirilmek istendiğinde (modelleri hareket ettirmek, belirli koşullar sağlandığında yeni model eklemek vb.), Gazebo için hızlı bir arayüz istendiğinde, oluşturulan eklenti başkaları için de faydalı olabilecek şekilde paylaşılacak istendiğinde eklentiler kullanılabilir.

Mevcut olarak altı tane eklenti türü bulunmaktadır.

- Dünya
- Model
- Sensör
- Sistem
- Görsel
- Arayüz



Eklentiler (Plugins)

Her eklenti türü farklı bir Gazebo bileşeni tarafından yönetilmektedir. Örneğin, bir model eklentisi Gazebo'daki belirli bir modele eklenmekte ve bu modeli kontrol etmektedir.

Eklenti türü, istenilen işlevselliğe göre seçilmelidir. Örneğin;

- Fizik motoru, ortam aydınlatması vb. dünya özelliklerini kontrol etmek için bir dünya eklentisi,
- Eklemlerin ve modelin durumunu kontrol etmek için bir model eklentisi,
- Sensör bilgilerini elde etmek ve sensör özelliklerini kontrol etmek için ise bir sensör eklentisi

kullanılabilir.



Eklentiler (Plugins)

Eklentiler, aşağıdaki şekilde terminal komut satırından yüklenebileceği gibi;

```
gzserver -s <plugin_filename>
```

SDF dosyası içerisinde aşağıdaki şekilde belirtilerek de;

```
<plugin name="gazebo_ros_control" filename="libgazebo_ros_control.so">  
  <robotNamespace>/MYROBOT</robotNamespace>  
</plugin>
```

yüklenebilirler.



Gazebo Kurulumu

Gazebo, ROS tarafından kullanılan bağımsız bir projedir.

Genellikle, her ROS sürüm döngüsünün başında mevcut olan en yeni Gazebo sürümü tam olarak entegre ve desteklenecek şekilde resmi olarak seçilmektedir.



Gazebo Kurulumu

Ubuntu üzerinde Gazebo yüklemek için paketler kullanılabilir. Gazebo paketlerine ev sahipliği yapan iki ana kaynak (packages.ros.org ve packages.osrfoundation.org) bulunmaktadır.

- packages.ros.org üzerinde
 - ROS Indigo: Gazebo 2.x
 - ROS Kinetic: Gazebo 7.x
 - ROS Lunar: Gazebo 7.x
 - ROS Melodic: Gazebo 9.x
- packages.osrfoundation.org üzerinde
 - gazebo 7.x (gazebo7 paket adıyla)
 - gazebo 8.x (gazebo8 paket adıyla)
 - gazebo 9.x (gazebo9 paket adıyla)

paketleri yer almaktadır. Bu kaynaklardan biri, Gazebo yüklemek için kullanılabilir.



GAZEBO

Gazebo Kurulumu

ROS'un belirli bir sürümünü çalıştırması gereken ve Gazebo ROS ile ilgili tüm paketleri kullanıma hazır şekilde kullanmak isteyen kullanıcılar için, packages.ros.org kaynağında yer alan Gazebo sürümünün kullanılması önerilmektedir. Örneğin ROS Kinetic, Gazebo'nun 7.x sürümünü barındırmakta ve kullanmaktadır.



Gazebo Kurulumu

İhtiyaç duyulduğu takdirde önerilen seçeneklerin dışında belirli bir Gazebo ve ROS versiyonunu birlikte kullanmak mümkündür.

Ancak bu durumda Gazebo ile ilgili herhangi bir ROS Ubuntu paketi, ROS dağıtım kaynağından kullanılamamaktadır.

Paketlerin eşdeğeri OSRF kaynağından yüklenebilir, fakat diğer bütün yazılımlar catkin_workspaces kullanılarak kaynaktan oluşturulmalıdır.

Ayrıca mevcut ROS kurulumuyla beraber gelen bir Gazebo versiyonu mevcutsa, farklı bir Gazebo versiyonu kurulmak istendiğinde ilk olarak bu versiyon kurulumunun kaldırılması gerekebilir.



Gazebo Çalıştırılması

Terminalden

`gazebo`

komutu çalıştırarak Gazebo başlatılabilir.

Grafik arayüzünde de görülebileceği üzere, Gazebo istemcisi sayesinde çalışan simülasyonu belirli bir ölçüde modifiye etmek mümkündür.



Çalışma Ortamının Hazırlanması

ROS-Gazebo çalışmalarını gerçekleştirebilmek adına Catkin çalışma ortamı hazırlanabilir.



Dünya Oluşturulması

Simülasyon ortamında kullanılacak dünyayı oluştururken, Gazebo veritabanında yer alan hazır dünya dosyalarından yararlanılabildiği gibi, kullanıcı kendi dünyasını da oluşturabilmektedir.

Dünya dosyası;

- grafik arayüzü aracılığıyla,
- elle SDF formatına uygun tanımlamalar yaparak,

oluşturulabilir.



Model Oluşturulması

Simülasyon ortamında kullanılacak modeller için dünya oluştururken olduğu gibi model dosyası da;

- grafik arayüzünün içerdiği “Model Editor” aracılığıyla,
- elle SDF veya URDF, URDF.XACRO formatına uygun tanımlamalar yaparak,

oluşturulabilir.

SDF formatı kullanılarak oluşturulan dosyalar doğrudan Gazebo ile kullanılabilirken, URDF ve URDF.XACRO formatındaki dosyalar ekstra işleme gerek duyabilmektedir.



Mesh Giydirmeye

Modeller oluşturulurken `<visual>` etiketinde bir model dosyası tanımlanarak model üzerine kalıp giydirilebilir.

`<visual>` etiketinde tanımlama yapıldığı takdirde, bu durumdan modelin sadece görsel özelliği etkilenmektedir.



Sensör Ekleme

Modeller oluşturulurken, model dosyasında <link>, <joint> ve gerektiğinde .urdf dosyaları için <gazebo reference> tanımlamaları yaparak modele sensör eklenebilmektedir.



ROS ile Kontrol

Bir modeli simülasyon ortamında ROS ile kontrol edebilmek için öncelikle ilgili paketlerin (gazebo_ros_pkgs) kurulu olması gerekmektedir.

Catkin ortamı oluşturulduğunda ve ROS ile Gazebo kurulumları ve yapılandırması uygun bir şekilde gerçekleştirildiğinde

```
roslaunch gazebo_ros gazebo
```

komutu başarılı bir şekilde Gazebo'yu ROS entegrasyonu ile beraber çalıştıracaktır.

