

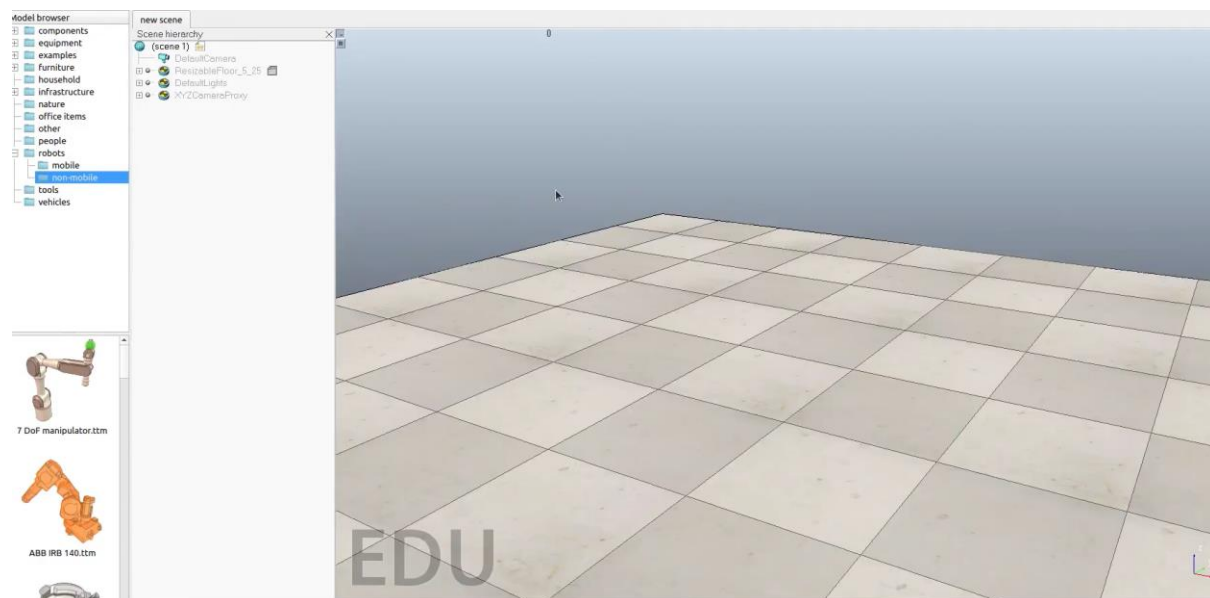
Hafizh Alfian S

1103201260

Chapter 5

Konfigurasi CoppeliaSim

Untuk memulai komunikasi antara ROS dan simulasi, kita harus menginstalnya di sistem kita dan mengonfigurasi lingkungan. CoppeliaSim, yang dikembangkan oleh Coppelia Robotics GmbH, adalah perangkat lunak lintas platform yang dapat digunakan untuk sistem operasi seperti Windows, macOS, dan Linux. Perangkat lunak ini tersedia secara gratis sebagai lisensi pendidikan dan sebagai lisensi komersial. Untuk mendapatkan simulator CoppeliaSim versi terbaru, kunjungi halaman unduhan Coppelia Robotics di sini: <http://www.coppeliarobotics.com/downloads.html>, dan pilih versi edu untuk Linux. Bab ini akan membahas versi CoppeliaSim 4.2.0.



Daftar antarmuka pemrograman aplikasi jarak jauh (Remote API): Remote API CoppeliaSim terdiri dari beberapa fungsi yang dapat digunakan oleh aplikasi eksternal yang dikembangkan dalam bahasa C/C++, Python, Lua, atau MATLAB. Remote API berinteraksi dengan CoppeliaSim melalui jaringan melalui komunikasi socket. Anda juga dapat mengintegrasikan remote API ke dalam node C++ atau Python Anda untuk menghubungkan ROS ke adegan simulasi. Jika Anda ingin menggunakan API remot, Anda harus mengimplementasikan sisi klien dan server seperti yang ditunjukkan di bawah ini:

A. Klien CoppeliaSim: Sisi klien terletak di aplikasi eksternal. Ini dapat dibuat dalam node ROS atau dalam program standar yang ditulis dalam salah satu bahasa pemrograman yang didukung.

B. Server CoppeliaSim: Sisi ini terletak dalam skrip CoppeliaSim dan memungkinkan simulator menerima data dari sumber eksternal untuk berinteraksi dengan adegan simulasi.

RosInterface: Antarmuka yang saat ini memungkinkan komunikasi antara CoppeliaSim dan ROS. Sebelumnya, plugin ROS digunakan, tetapi sekarang tidak disarankan lagi.

Webots

Sebelum mengonfigurasi Webots dengan ROS, kita harus menginstal Webots di sistem kita, seperti yang dilakukan dengan CoppeliaSim. Webots awalnya dikembangkan oleh Swiss Federal Institute of Technology, Lausanne (EPFL), dan dapat digunakan di berbagai platform, termasuk Windows, Linux, dan macOS. Sekarang dikembangkan oleh Cyberbotics, dan tersedia di bawah lisensi gratis dan sumber terbuka Apache 2. Untuk memodelkan, memprogram, dan mensimulasikan robot, Webots menyediakan lingkungan pengembangan lengkap. Perangkat lunak ini banyak digunakan dalam industri, pendidikan, dan riset karena dirancang untuk penggunaan profesional.

Kesimpulan

Bab ini sebagian besar mengulangi apa yang telah kita lakukan di bab sebelumnya dengan Gazebo, menggunakan simulator robot CoppeliaSim dan Webots. Kedua program simulasi multiplatform ini sangat serbaguna dan menggabungkan berbagai teknologi. Mereka mungkin lebih mudah digunakan bagi pengguna baru berkat antarmuka pengguna yang mudah digunakan.

Kita mensimulasikan dua robot secara khusus. Yang pertama diimpor menggunakan file URDF dari lengan robot tujuh-DOF yang telah dirancang di bab-bab sebelumnya. Robot kedua adalah robot roda-diferensial yang populer yang disediakan oleh model simulasi Webots. Kita belajar bagaimana menghubungkan dan mengontrol sendi robot pada model kita dengan ROS dan menggunakan topik untuk menggerakkan robot mobile.

Pertanyaan di buku

- Cara CoppeliaSim dan ROS berkomunikasi:

CoppeliaSim dan ROS dapat berkomunikasi menggunakan antarmuka yang disebut sebagai RosInterface. Antarmuka ini memungkinkan CoppeliaSim untuk bertindak sebagai node ROS yang dapat menerima perintah dan berbagi data dengan ROS.

- Cara mengontrol simulasi CoppeliaSim dengan ROS:

Kontrol simulasi CoppeliaSim dengan ROS dapat dilakukan melalui RosInterface. Dengan RosInterface, kita dapat menggunakan node ROS untuk mengirim perintah ke CoppeliaSim, seperti mengontrol gerakan robot atau memperoleh data dari simulasi.

- Cara mengimpor model robot baru ke CoppeliaSim dan mengintegrasikannya dengan ROS:

Untuk mengimpor model robot baru ke CoppeliaSim, kita dapat membuat deskripsi robot dalam format seperti URDF atau file lain yang didukung oleh CoppeliaSim. Setelah itu, kita dapat mengintegrasikan model tersebut dengan ROS menggunakan RosInterface untuk berkomunikasi antara robot dalam simulasi dengan node ROS.

- Apakah Webots dapat digunakan sebagai perangkat lunak mandiri:

Ya, Webots dapat digunakan sebagai perangkat lunak mandiri untuk memodelkan, memprogram, dan mensimulasikan robot tanpa ketergantungan pada ROS atau perangkat lunak lainnya.

- Cara komunikasi antara ROS dan Webots:

Komunikasi antara ROS dan Webots dapat dilakukan dengan RosInterface yang disediakan oleh Webots. RosInterface memungkinkan Webots bertindak sebagai node ROS, memungkinkan pertukaran data antara lingkungan simulasi Webots dan node-node ROS lainnya.