

Agent-based Multi Robot System

User Manual

Release version 1.0

Introduction

This document describes the developer's technical configuration for setting up the important components, the necessary steps for running the system and notes for existing issues and complexities.

Machine Configuration

The first important thing to note is that the developer created this program on an old Samsung personal computer (PC) – system model: 340XAA/350XAA/550XAA - with the following resources:

- Intel® Core™ i3-7020U CPU @ 2.30GHz
- 2 cores and 4 logical processors
- 8GB of installed RAM
- Windows 10 64-bit Operating System (OS), x64-based processor

The system was written using ROS1 and not ROS2, therefore a compatible OS is required. You may use the virtual machine software of your choice, however, in this case VMware was used so please refer to external guides and tutorials for the software of your choice if you find details in this document to be not relevant or applicable. The specific environment used was the VMware Workstation 16 player (non-commercial use) [version 16.2.x] with the following configuration.

- 6GB memory allocated
- 4 logical processor cores
- 33GB maximum hard disk space
- Ubuntu 20.04

If you find that you are unable to allocate these resources or don't have the sufficient hardware on your machine, then it's very unlikely that you would be able to run this program. Please consider the current configuration as the minimum requirements possible to run this program.

Complications

During the initial setup, the original VM configuration used was 4GB of allocated memory and 2GB for accelerating 3D graphics. For a machine without a dedicated graphics card this setup causes the Gazebo simulation software to crash after one or two minutes. One fix is to turn off 3D graphics acceleration and allocate more memory to the configuration.

Several Java versions are not compatible with the Jason software. The developer ended up using Java SE 15 and installed the packages for dependencies of java-roslaunch.

Process

1. Once you have a suitable machine setup, whether you are using a VM or not, install the Noetic distribution for ROS, for Ubuntu [<https://wiki.ros.org/noetic/Installation/Ubuntu>]. It is necessary to install the full desktop version in order to obtain packages for simulators and perception.
2. Install the rosbridge package [http://wiki.ros.org/rosbridge_suite].
3. You may wish to install whatever IDE you want for programming in Python. The developer used Geany v4.2.
4. Install Java SE 15 [<https://www.oracle.com/java/technologies/javase/jdk15-archive-downloads.html>]. Keep in mind that Oracle has changed their license and archived this version of Java. They are not up-to-date with security patches and should not be used for production.
5. Install the latest version of the JEdit text editor [<http://www.jedit.org/index.php?page=download>].
6. Install the Jason software by following this tutorial [<http://jason.sourceforge.net/mini-tutorial/getting-started/>].
7. In order to run the system the first step is to run ROS as you would normally, then the rosbridge server with the command:
`roslaunch rosbridge_server rosbridge_websocket.launch`
8. Note that within the interface environment, in the initialisation that is called prior to MAS execution, the default URI for ROS bridge is 9090. The command line argument taken is:
`ws://localhost:9090`
9. Run the multi-agent system in JEdit, the MAS console should print out the information confirming your connection to ROS bridge.
10. Run the robot system using the environment's launch file:
`roslaunch [package name] world_env.launch`