Universidade da Beira Interior
Departamento de Informática
Inteligência Artificial

Practical Project

2019-20

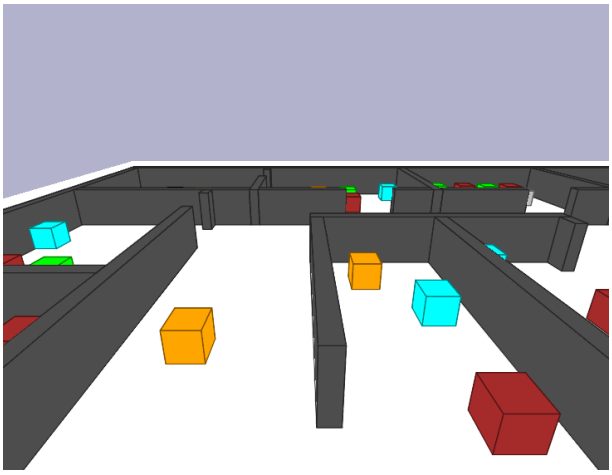
Introduction

This project is about the development of a robotic agent that travels in a world, finds objects and answers questions that are asked by the user. This world is a floor in a hotel.

The robot is inside a simulator called Stage, and Stage works in ROS, which is the Robot Operating System. You have received together with this text, a virtual machine that has ROS already installed.

If you run the command `./go.sh` in a terminal, it will open the Stage and allow you to drive the robot inside the world.

Two views of the environment are presented below.



The 2D view shows the complete world and there are several colored squares that represent objects in the world, that also includes the robot represented by a dark blue small cube.

Project

Setup

What you have to do is complete a node, that has already some functionality, called `agent`. It receives information from other nodes. This information should be represented in a way that you consider adequate. The code for this node is in the file `/home/viki/catkin_ws/src/ia/src/agent.py`. Do not move it to another location. Once you change the node you can make it run again by closing the window that contains several terminals, and giving the `./go.sh` command again in a terminal.

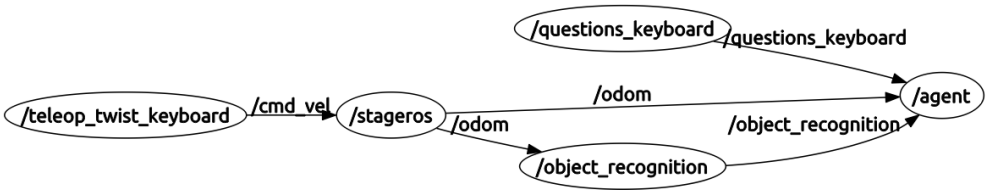
The robot will be traveling in the world, and its movement is controlled by the user using the keyboard.

During its path it will see several objects. When it approaches them, the name of these objects and their categories (see below) is published as a message by the `object_recognition` node. So your node subscribes to this node to receive these messages.

There are other messages that are of interest to you:

- `odom`: it gives the coordinates of your position in the world;
- `questions_keyboard`: gives a number that corresponds to a question asked by the user. Your agent has to answer these questions.

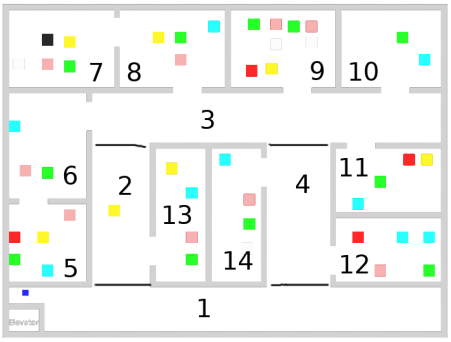
The following graph shows the existing nodes and the messages that they send to each other:



How the world works

Your robot will always start in a corridor, in front of the elevators.

You can infer the number of the room you are in at a given time using the coordinates of the robot and the following room number scheme:



Note that the numbers from 1 to 4 are assigned to corridors and not to rooms.

The books were distributed randomly in the hotel. The tables were placed more frequently in rooms that already had books. Chairs are more likely to be in rooms that have tables.

Possible **categories** are the following: bed, book, chair, computer, door, person, table. Note that other categories might be present. The categories names' are always a single word.

Individual **objects** receive names that always include the category first, such as: person_joe, book_alice_in_wonderland, table_table1, computer_windows2.

There are five **room types**, all of them can have tables and chairs:

- if the room contains one individual bed, it is a single room;
- if the room contains two individual beds, it is a double room;
- if there are beds and an internal door (that connects to another room) then it is a suite;
- if the room contains only one table and several chairs it is a meeting room
- rooms that are none of the above are called generic rooms

We say that a room is **occupied** if there is at least one person in it.

Note that your code will be tested with different versions of the world where everything can change (positions of objects, number of objects, number of categories). The only thing that remains is the walls, but the doors might also change.

Questions

Your robot must be able to answer these questions, at any point in time:

1. How many rooms are not occupied?
2. How many suites did you find until now?
3. Is it more probable to find people in the corridors or inside the rooms?
4. If you want to find a computer, to which type of room do you go to?
5. What is the number of the closest single room?
6. How can you go from the current room to the elevator?
7. How many books do you estimate to find in the next 2 minutes?
8. What is the probability of finding a table in a room without books but that has at least one chair?

Extra points

Your work can earn extra points if it goes beyond what is asked here in the sense that it presents more functionality.

This can be whatever you can think, from visualization, to better user interface, to answering more interesting questions (not simple variations on the existing questions).

Please make clear in the report if you included extra functionality in your code and carefully explain what it is and how it works.

Presentation

Your group will make a presentation of your work with 5 minutes duration. Do not spend this time talking about things that are already in this description: focus on what you implemented and how it works. Also talk about what were the difficulties your had and how you were able to solve them. Both group members have to talk: ideally each will use half of the presenting time. After the presentation there is a 5 minute period for questions.

Deliverables

You have to deliver the following:

- a PDF report explaining your work. The report must explain clearly which part of the work was done by which group member. It is not acceptable for you to write something like "this part of the work was done together".
- the PDF slides of your presentation (not PPT, not online).
- the source code for your ROS node. It must run in a virtual machine identical to the one you received with this text. It must be well documented (code comments).

The deliverables must be submitted in a single zip file with both your student numbers in the file name, by email to `luis.alexandre@ubi.pt` until midnight of 2020-01-02. You **must** to submit by this date. If you do not submit whatever you have, even if it is not "finished" yet, you will fail the Project and this course. Do not wait for the final day. We seriously advise you to submit, at most, on the day before the deadline to avoid any possible problem. All submissions will be acknowledged by an email. If you do not receive an email acknowledging the reception of your work then speak with the professor urgently. He has 24 hours to acknowledge the reception.

The final practical classes will be devoted to help you with the Project, but do not wait for them to start working.

Bibliography

Here are some pointers to help you with the Project:

- Theoretical class slides (check Moodle).
- You need VirtualBox to open the virtual machine: `https://www.virtualbox.org/`. Choose a 64bit version.
- Everything about ROS can be found here: `http://ros.org`
- Download the virtual machine with the complete setup in `http://www.di.ubi.pt/~lfbaa/ia_vm2019.zip` . It is a 4GByte file.