

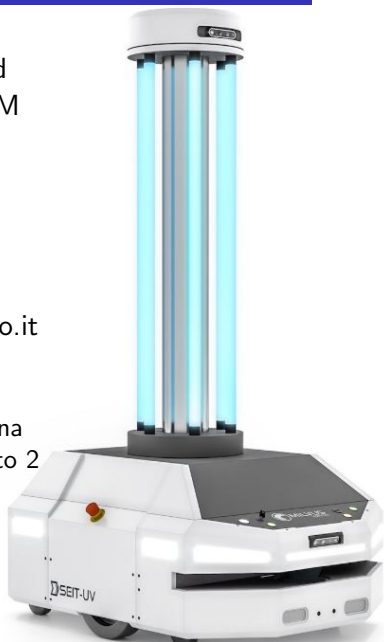
Final Project

Autonomous and
Mobile Robotics M

Gianluca Palli

gianluca.palli@unibo.it

DEI - LAR
University of Bologna
Viale del Risorgimento 2
40136 Bologna



Sanitizer Robot



Task 1

The students are asked to:

- setup the simulation of the TurtleBot3 burger robot using the Gazebo simulation environment and the TurtleBot3 Big House scenario. Note that during the exam evaluation, the robot can be spawned on the simulation scenario in a random position and additional obstacles can be present;
- OR
- setup the real TurtleBot3 robot and make it working in a confined environment with obstacles. Note that the shape and size of the environment will be not know in advance.

The students are asked to make the robot able to move autonomously in the environment to create a map covering the larger possible part of the environment:

- No previous knowledge of the map will be considered during the exam;
- Existing packages can be used (such as `explore_lite`);
- Implementation (or learning) of (sub)optimal map generation processes will be positively evaluated.

Task 3

After the mapping of the scenario (Task 2), the students are asked to make the robot able to navigate in the environment reaching a set of goals contained in a text file:

- The alignment between the map and the navigation system must be performed;
- SUGGESTION: to spread the `amcl` particles around in the environment, send an empty request to the `global_localization` service (`rosservice call /global_localization '{}'`);
- Implementation (or learning) of (sub)optimal map alignment processes will be positively evaluated.

Task 4

Suppose that:

- in order to kill the coronavirus, the robot is equipped with a set of UV lamps able to spread all around the robot a light power $P_l = 100 \mu\text{W m}^2$;
- the UV energy E at point (x, y) and time t can be computed as

$$E(x, y, t) = \int_0^t \frac{P_l}{(x - p_x(\tau))^2 + (y - p_y(\tau))^2} d\tau$$

where $p_x(t)$ and $p_y(t)$ represent the robot position along the x and the y axis respectively at time t ;

- any obstacle completely stop the UV power propagation;
- the light power emitted at a distance lower than 0.1 meters from the robot is zero due to the robot encumbrance;
- the room of interest can be discretized as a grid with resolution 0.2 m.

Task 4 (Cont.)

By sampling the process with sample time Δt , the UV energy E at point (x, y) and sample time k can be rewritten as

$$E(x, y, k) = \sum_{i=0}^k \frac{P_I \Delta t}{(x - p_x(i\Delta t))^2 + (y - p_y(i\Delta t))^2}$$

The students are asked to implement a strategy to define a path able to cover a selected room in the map in order to guarantee a minimum value of 10 mJ of the UV energy over the room to ensure the coronavirus is inactivated.

The students are asked to address the following tasks using ROS the TurtleBot3 robot moving in an unknown environment:

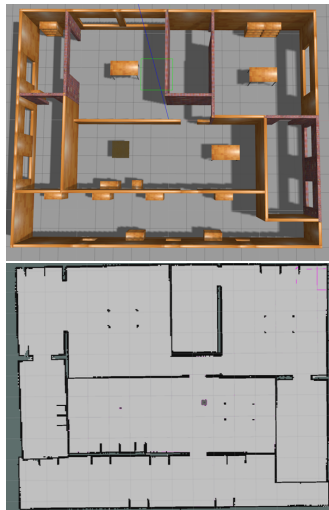
- The tasks can be developed either in simulation or with the real robot;
- A report (presentation or document) must be presented at the exam, together with the implemented code (provided separately, not in the report);
- During the exams (project presentation) the students will be asked to perform show the results by means of simulations, experiments with the real robot or videos;
- A clear organization of the project and the related code is highly recommended;
- Available ROS packages can be exploited for the project implementation.

Task 4 Implementation Example

ROS environment with simulation of
Turtlebot3 Burger robot running on
gazebo big house

Simulation services and actions:

- Lamps
- Energy control
- Task parameter update
- Sanification



Task 4 Implementation Example

Some rooms (or area) are defined through a table description

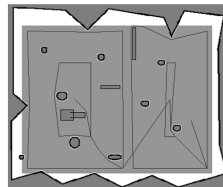
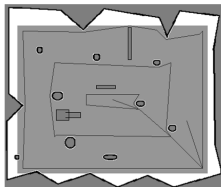
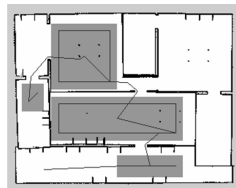
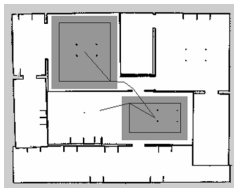


	ID	TOP LEFT	BOTTOM RIGHT
0	KITCHEN	[0, -4]	[4.5, -5,5]
1	OFFICE	[-4.5, 0]	[4.5, -3]
2	GARAGE	[-4.5, 5]	[0, 0.5]
3	GARDEN	[-6.5, 1]	[-5, -1]

Task 4 Implementation Example

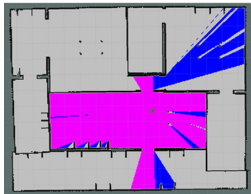
Path generation is divided in two steps:

- First, a path covering the room is defined on the base of the room shape
- Second, the path is adjusted to move around obstacles searching for the closest free point to the path in the obstacle neighborhood

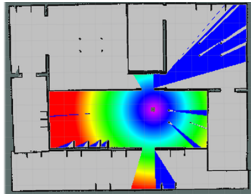


Task 4 Implementation Example

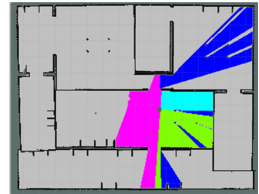
Grid maps are used to visualize the results



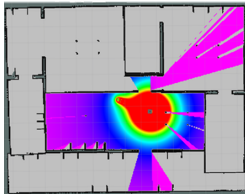
Visibility



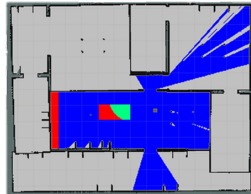
Distance



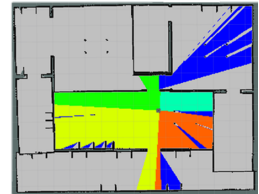
Lamp 1



Energy



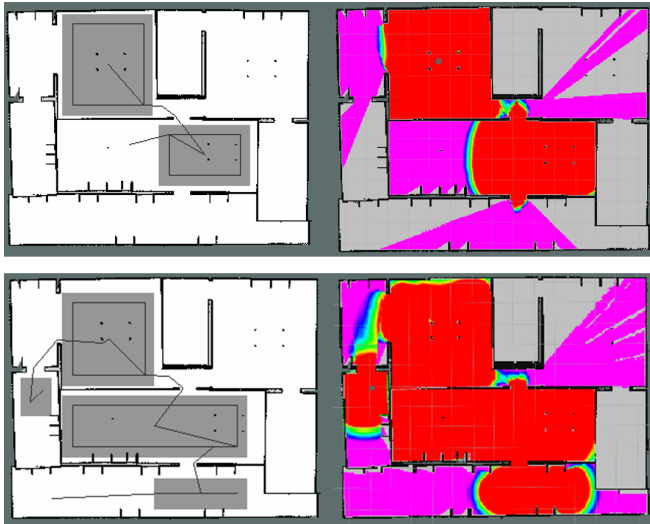
Control



Lamps 2

Task 4 Implementation Example

Results



Thanks!

Questions?