Multi-Robot Autonomous Exploration and Map Merging in Unknown Environments

Luiz Eugênio Santos Araújo Filho

July 9, 2021

Supervisor: Prof. Dr. Cairo L. Nascimento Jr.

Summary

Problem statement

- The problem known as Active Multi-Robot Simultaneous Localization and Mapping (Active MR-SLAM) states that multi-robots should perform autonomous exploration of unknown environments while mapping under pose uncertainty.

- Estimation of a global map representation of the environment via merging individual maps from different robots.

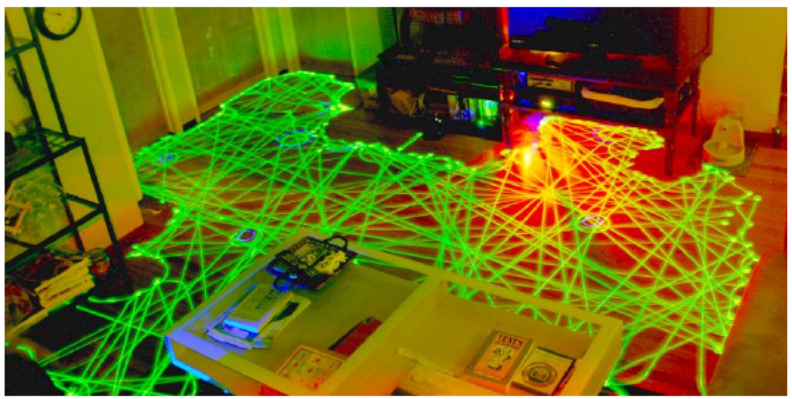
Problem statement

Autonomous exploration of multi-robots.

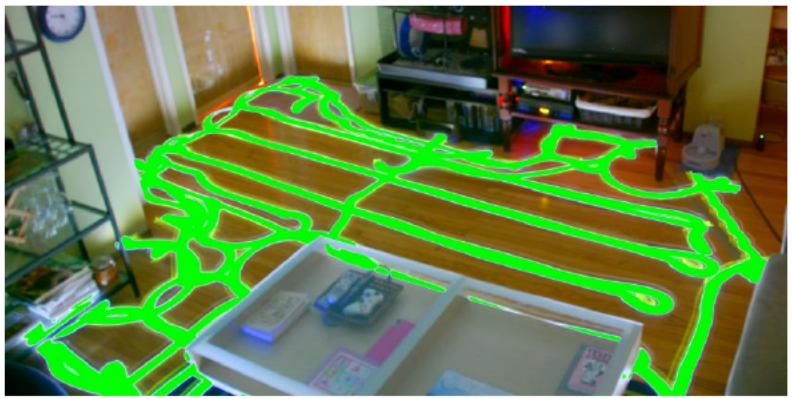
Cao Chao's Youtube channel.

Motivation

Autonomous robots have been slowly integrated into household products, such as vaccum cleaning robots.



Random navigation



Navigation aided by SLAM

Multi-Robot SLAM

Advantages of using multi-robot systems

- Cooperation has the potential to accomplish a task faster than a single robot
- Redundancy introduces fault-tolerant nature to the system
- Merging individual maps to compensate for global map uncertainty

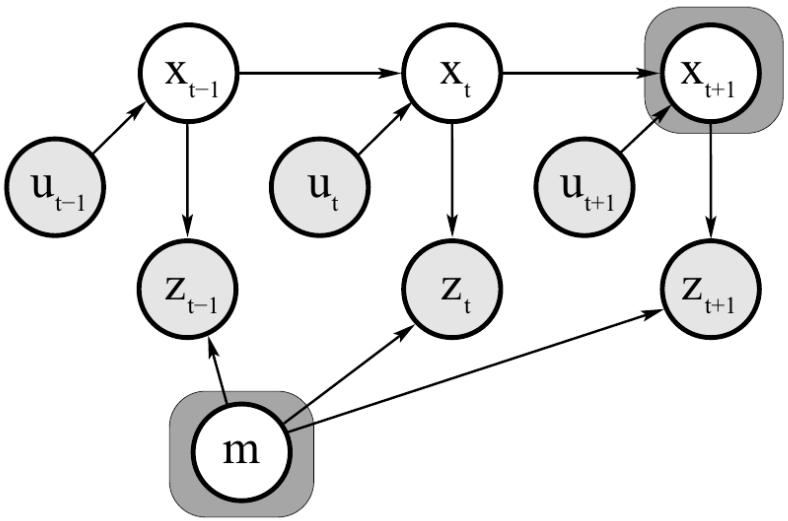
Multi-Robot SLAM

Difficulties when using multi-robot systems

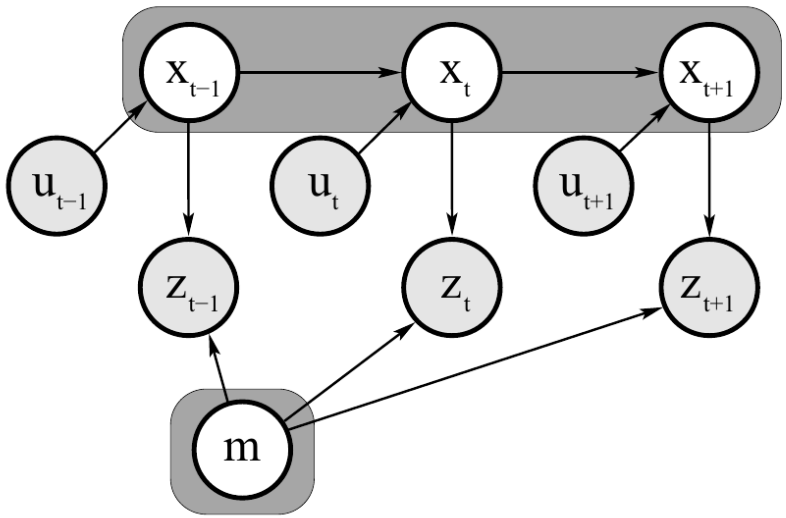
- Heterogeneous teams of robots
- Limited communication range and bandwidth
- Global map merging
- Relative pose computation
- Sensor interference (cross-talk)
- Latency and out-of-order package communication

Single-Robot SLAM

Online SLAM



Full SLAM



x_t: current robot pose

z_t: current sensor measurement

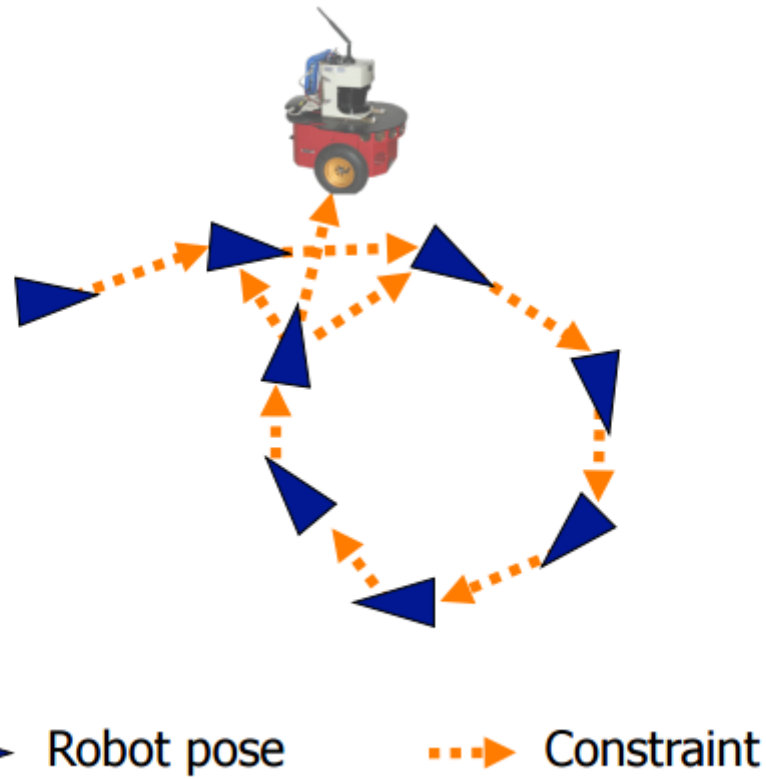
u_t: current input control

m: map representation of the robot environment

: objective of the estimation process

Single-Robot SLAM

GraphSLAM technique.

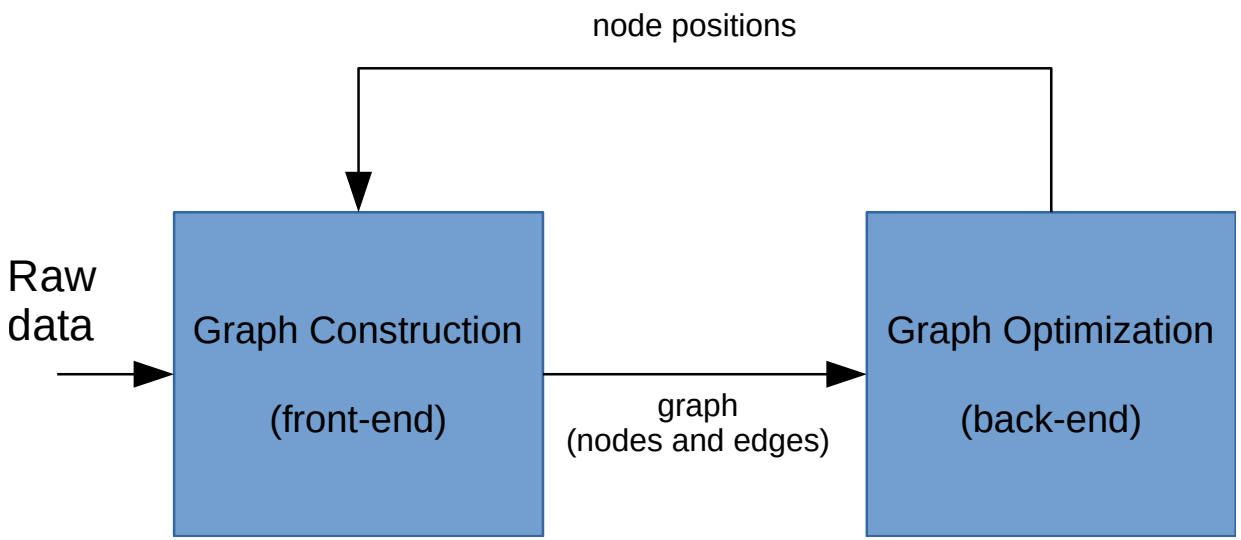


- Nodes are robot poses
- Edges are spatial constraints
- Graph is called a Pose graph

Single-Robot SLAM

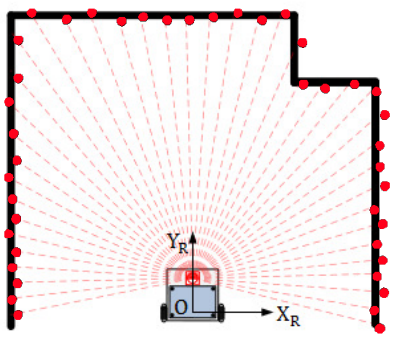
GraphSLAM technique.

- Graph is constructed by computing nodes relative transformations
- Optimization of pose graph is required to minimize the error introduced by constraints

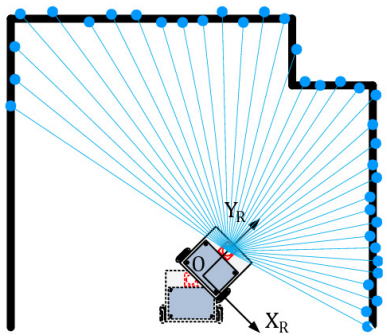


Single-Robot SLAM

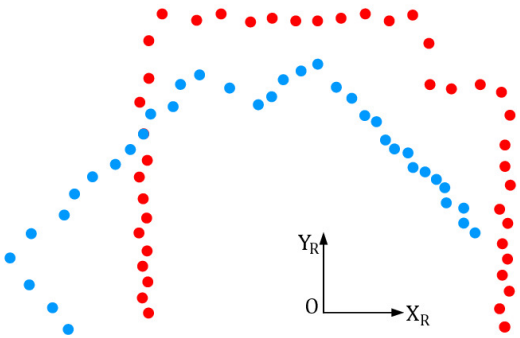
Scan matching of two consecutive poses.



Scan 1



Scan 2

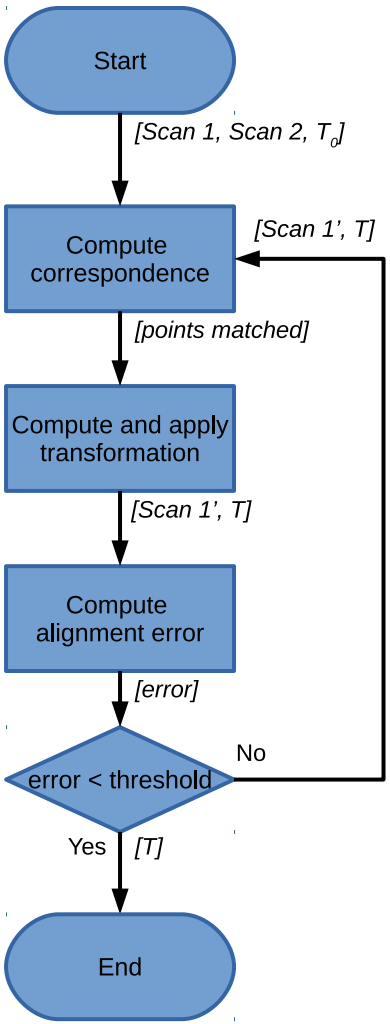


Scans overlapped

Scans matched

Single-Robot SLAM

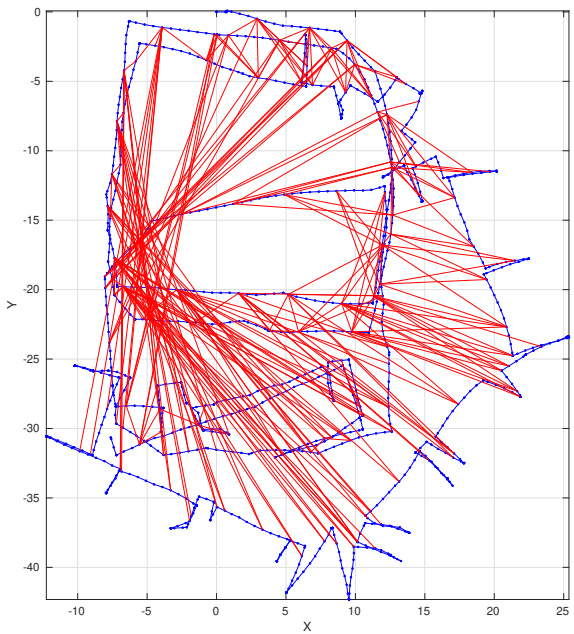
Iterative Closest Point algorithm (ICP).



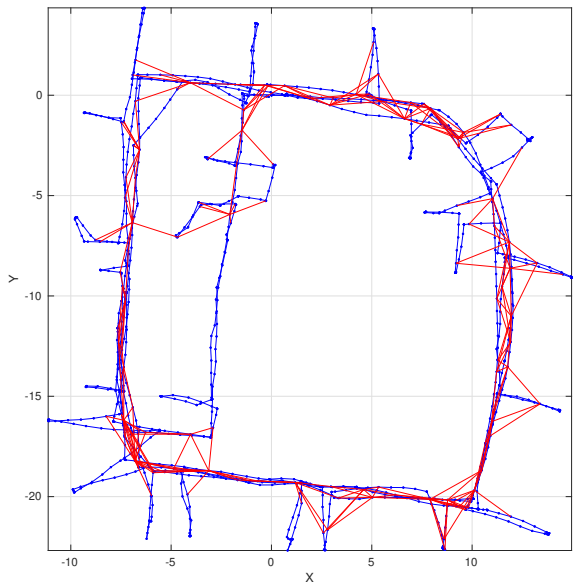
- T_0 is the transformation initial guess
- Point correspondence is found using nearest neighbors
- New transformation T is computed using Singular Value Decomposition
- Transformation is applied on Scan 1
- Error is computed using Scan 1' and Scan 2 points

Single-Robot SLAM

Pose graph optimization.



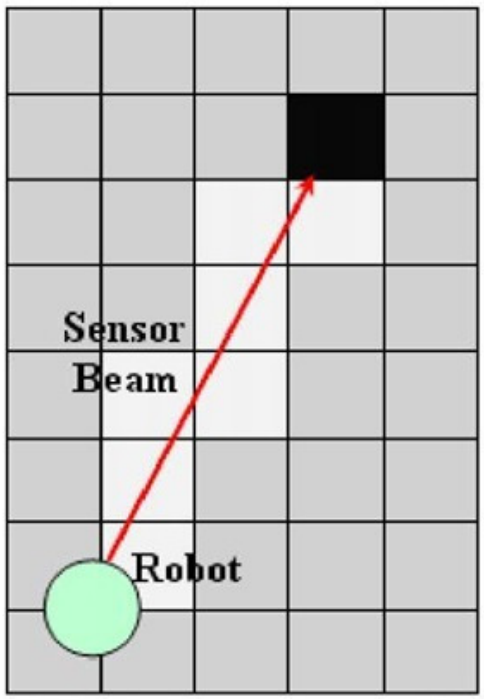
Before optimization



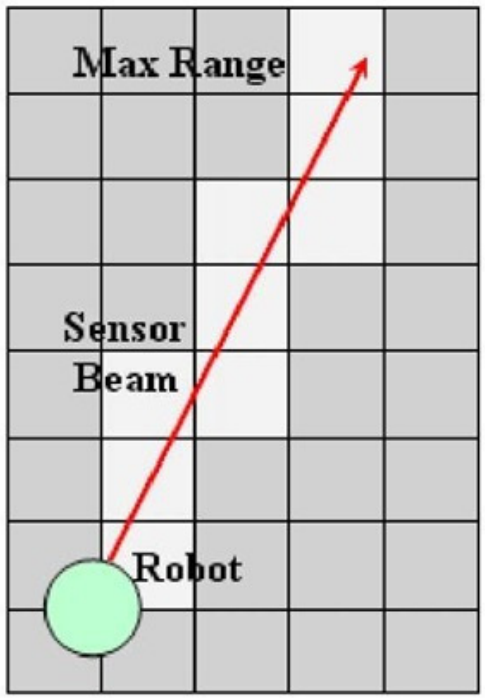
After optimization

Single-Robot SLAM

Occupancy grid mapping: gray level indicates the probability of cell occupancy (0 to 1).



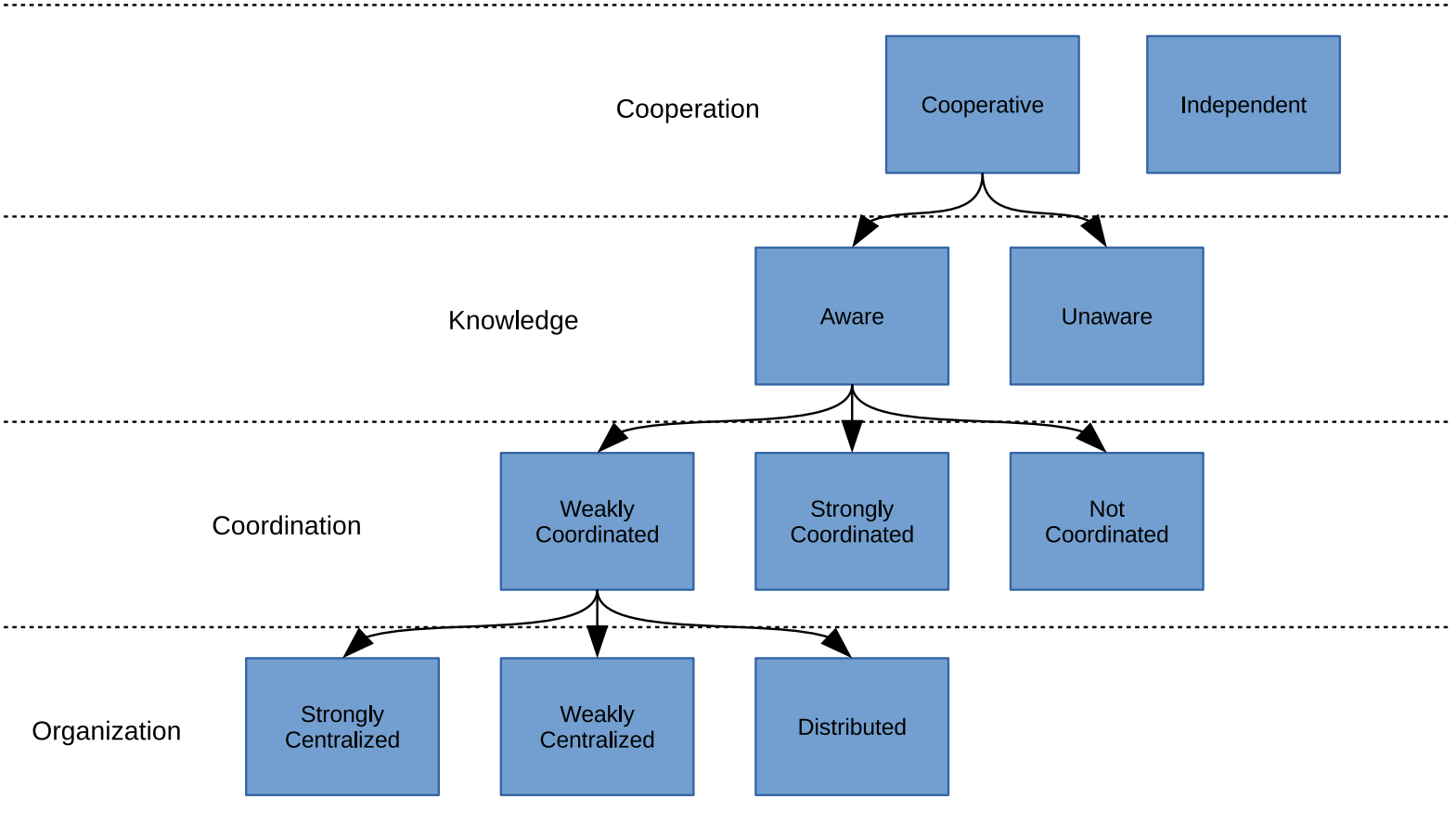
Obstacle detected



Obstacle not detected

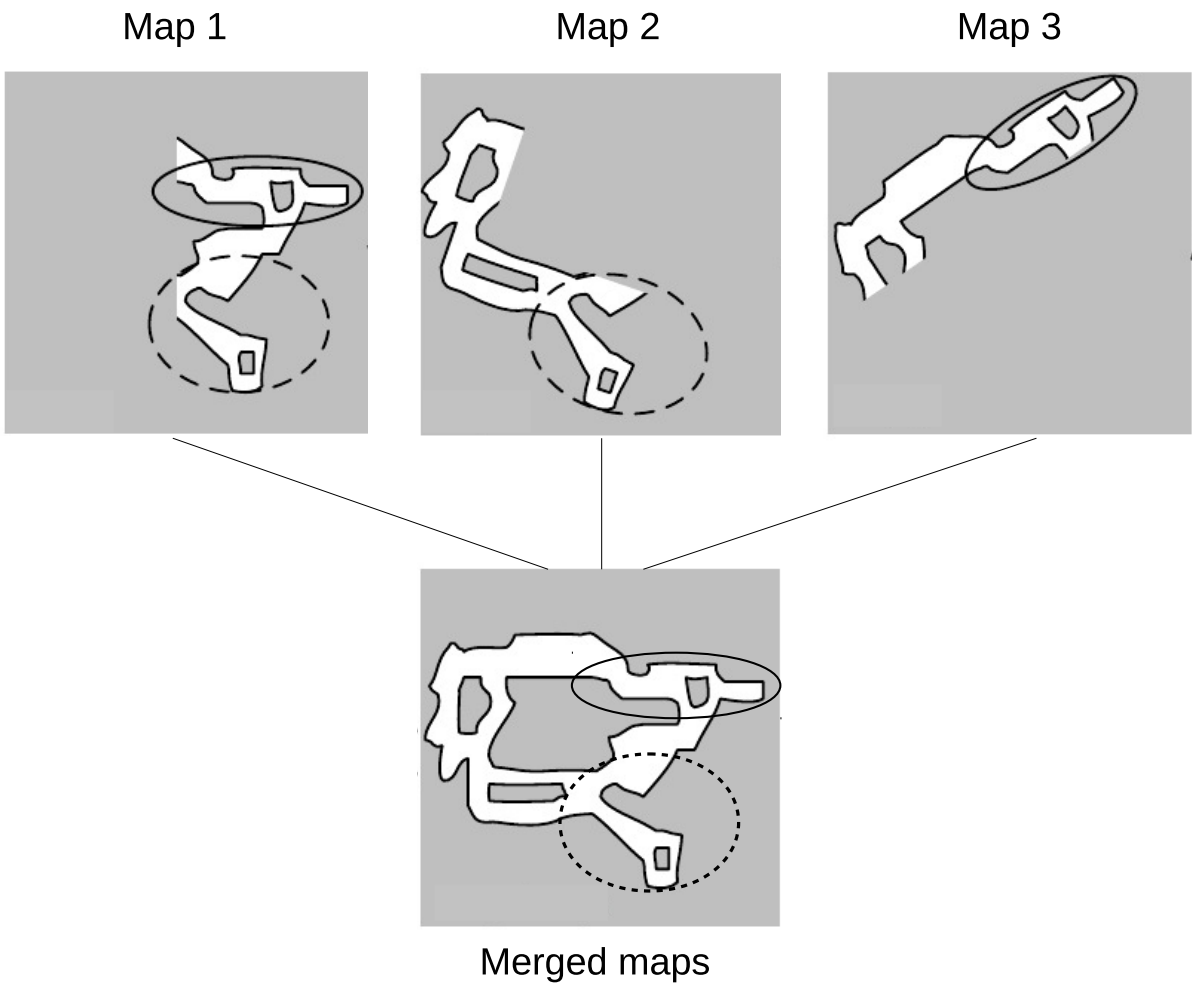
Multi-Robot SLAM

Levels of multi-robot cooperation.



Map merging

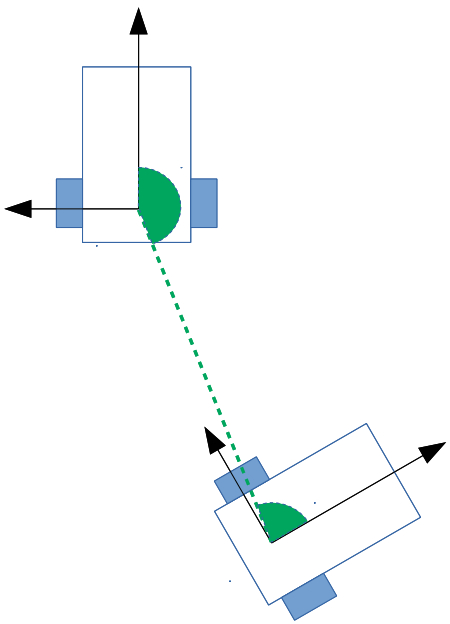
Map merging of 3 maps showing overlapped regions.



Map merging

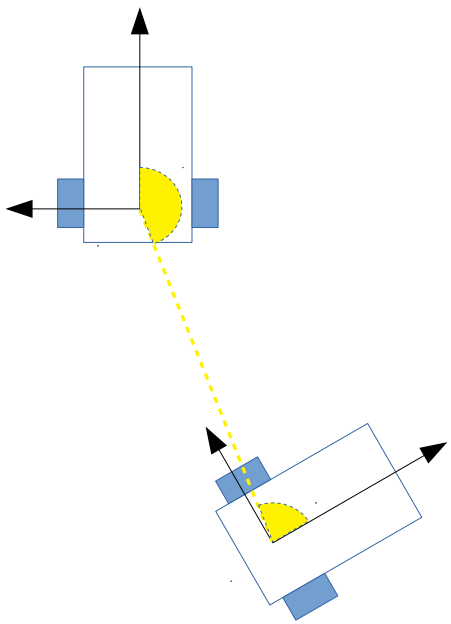
Scenarios for computing map merge.

Scenario 1



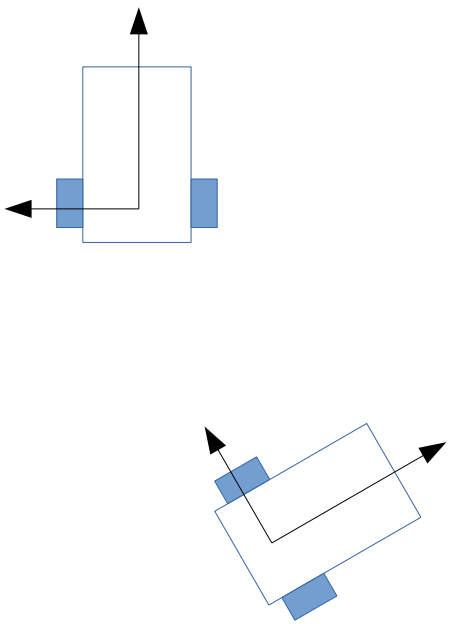
Relative poses are always known

Scenario 2



Rendezvous case

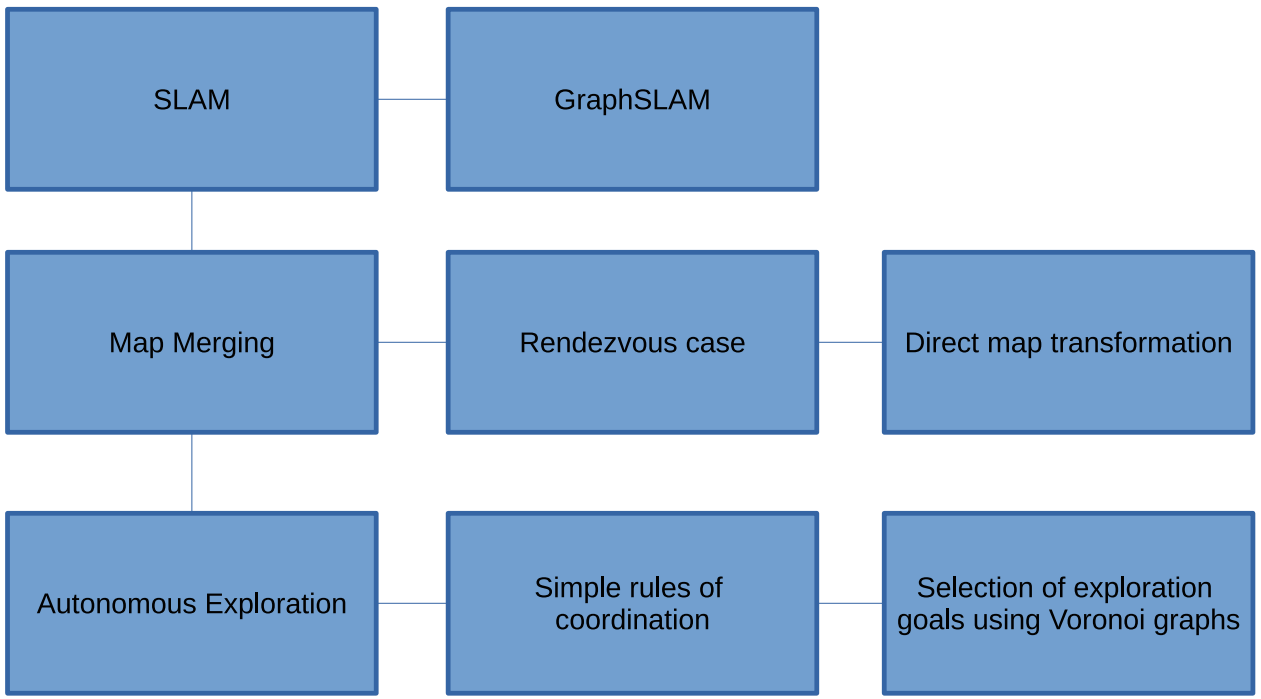
Scenario 3



Relative poses are never known

Active MR-SLAM

Simplified activity diagram.



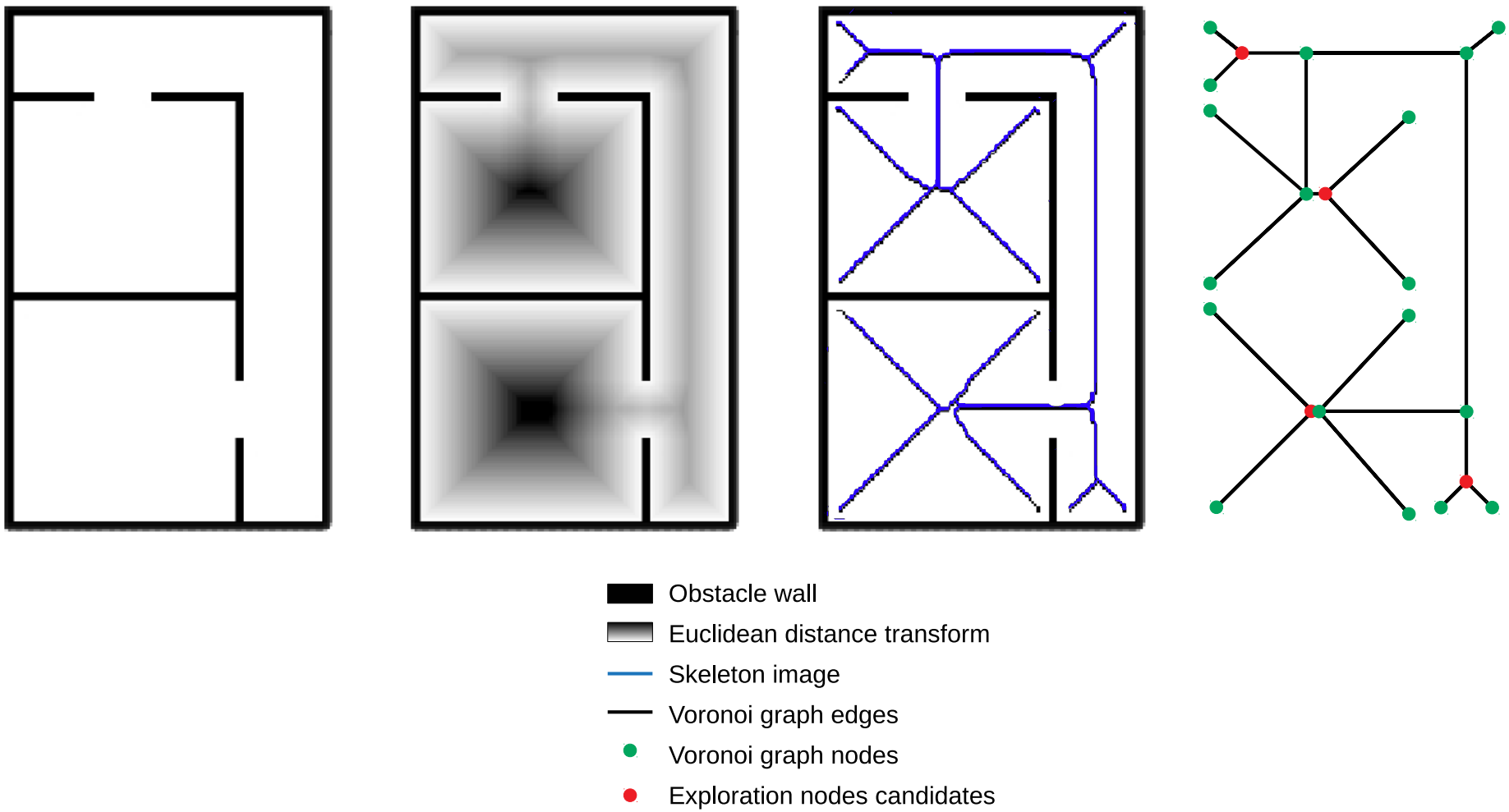
Active MR-SLAM

GraphSLAM architecture:

- ICP for scan matching
- Gauss-Newton method for pose graph optimization
- Aligned point cloud map
- Occupancy grid map

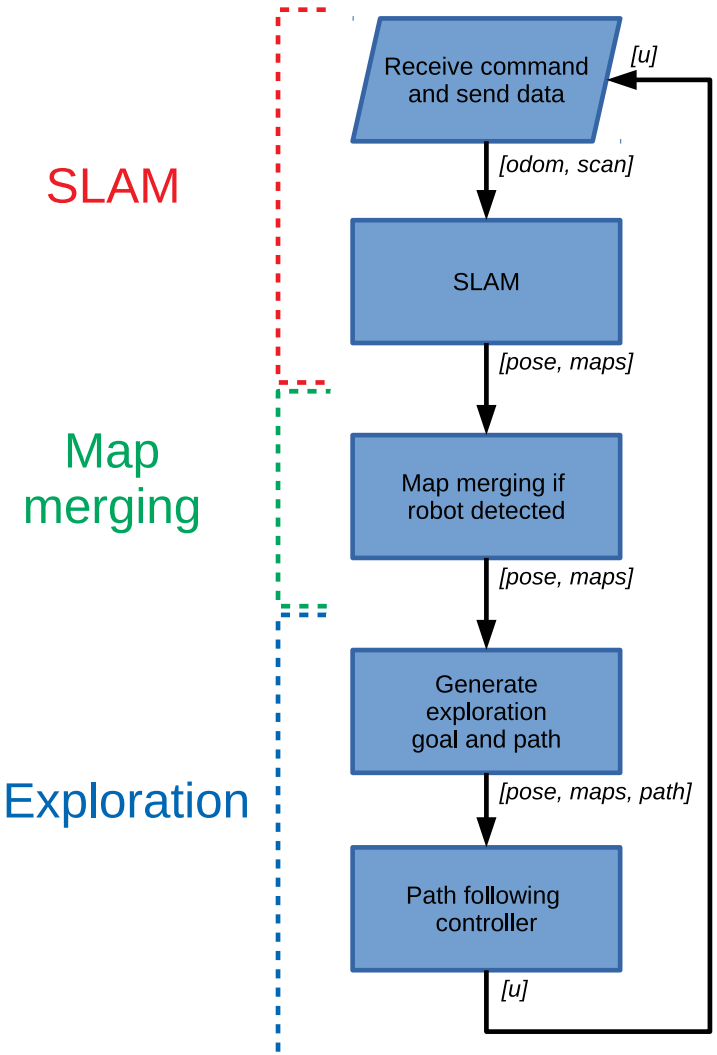
Active MR-SLAM

Extraction of Voronoi graphs from occupancy grid maps.



Active MR-SLAM

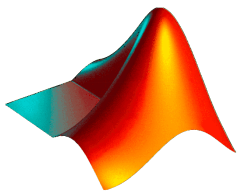
Active MR-SLAM simplified fluxogram.



- odom: wheel encoder measurements
- scan: scan measurements
- pose: current robot pose
- maps: point cloud and occupancy grid
- path: list of waypoints to exploration goal
- u: velocity control input vector

Setup for simulated and real world experiments

Software used for implementation of the proposed solution.



MATLAB®

GAZEBO

ROS

python

Setup for simulated and real world experiments

MATLAB:

- Active MR-SLAM algorithm.

ROS:

- Communication between MATLAB and the real/simulated robots.

Gazebo:

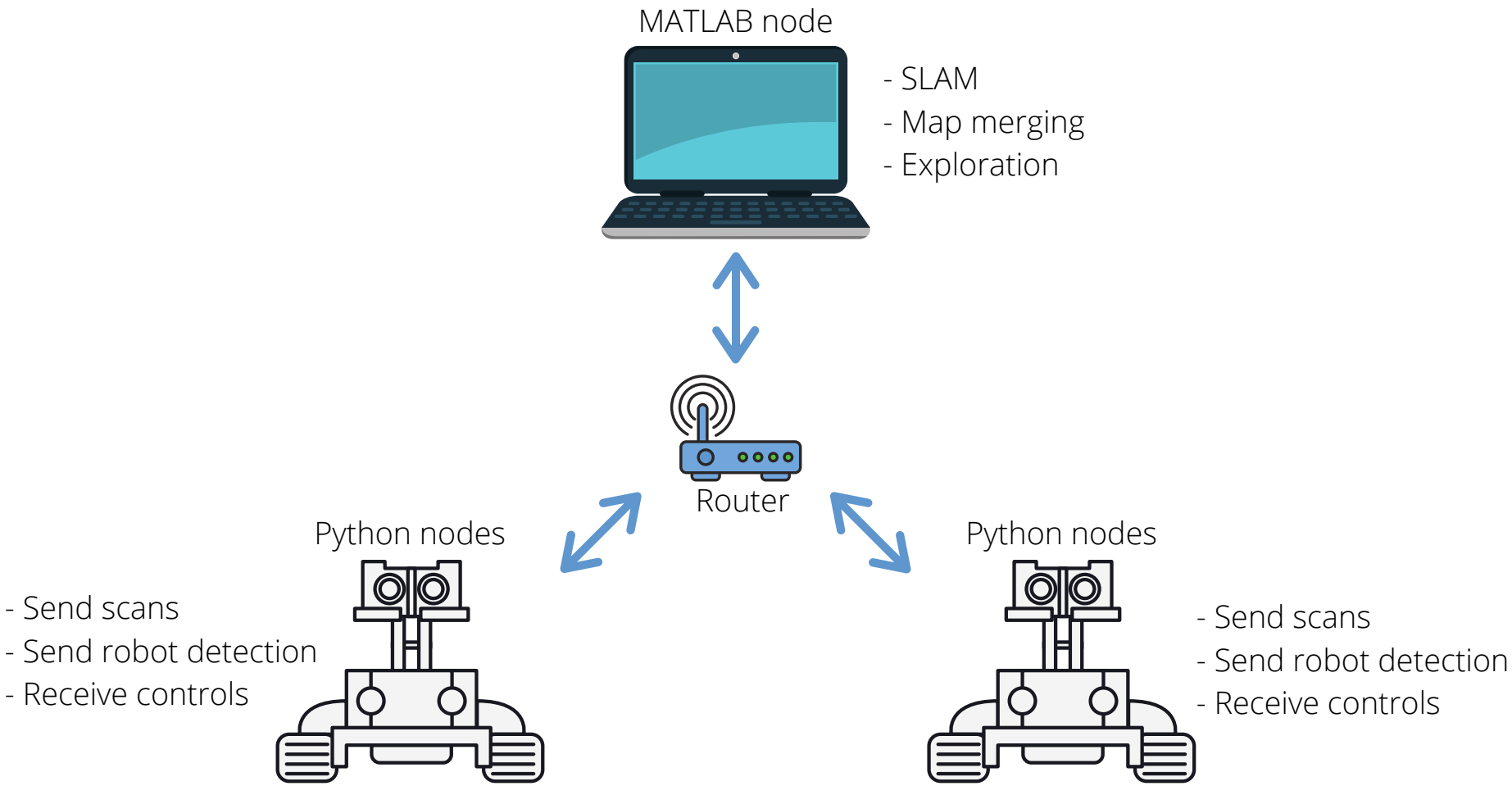
- Simulation of the environment and the multi-robot models.

Python:

- Programming language used for the low-level tasks executed by the processing boards embedded in the real robots.

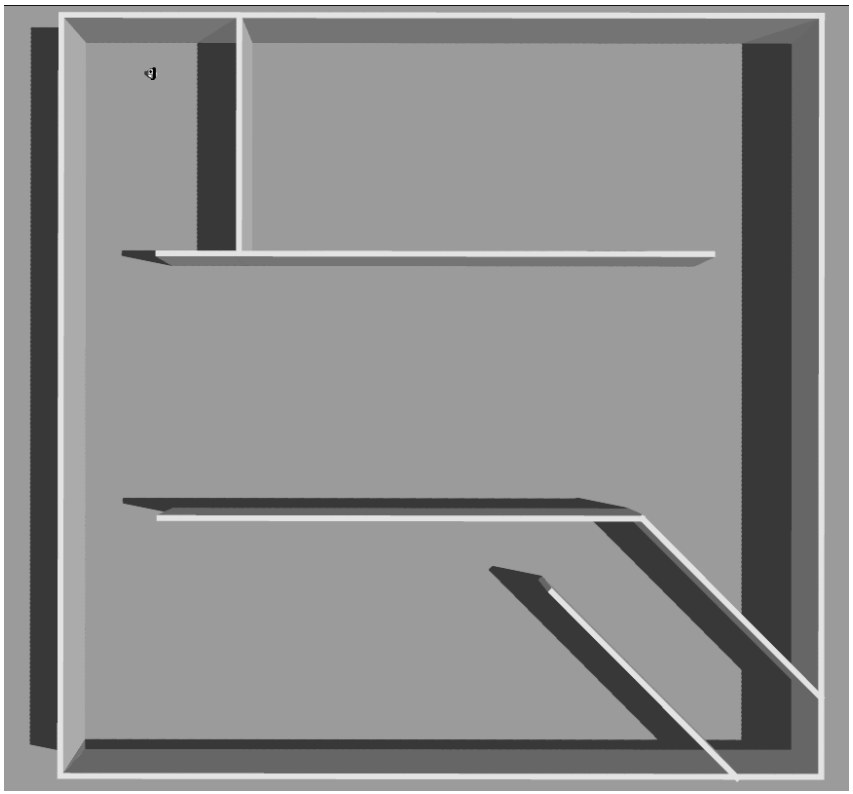
Setup for simulated and real world experiments

Communication between robots and central computer on ROS network.

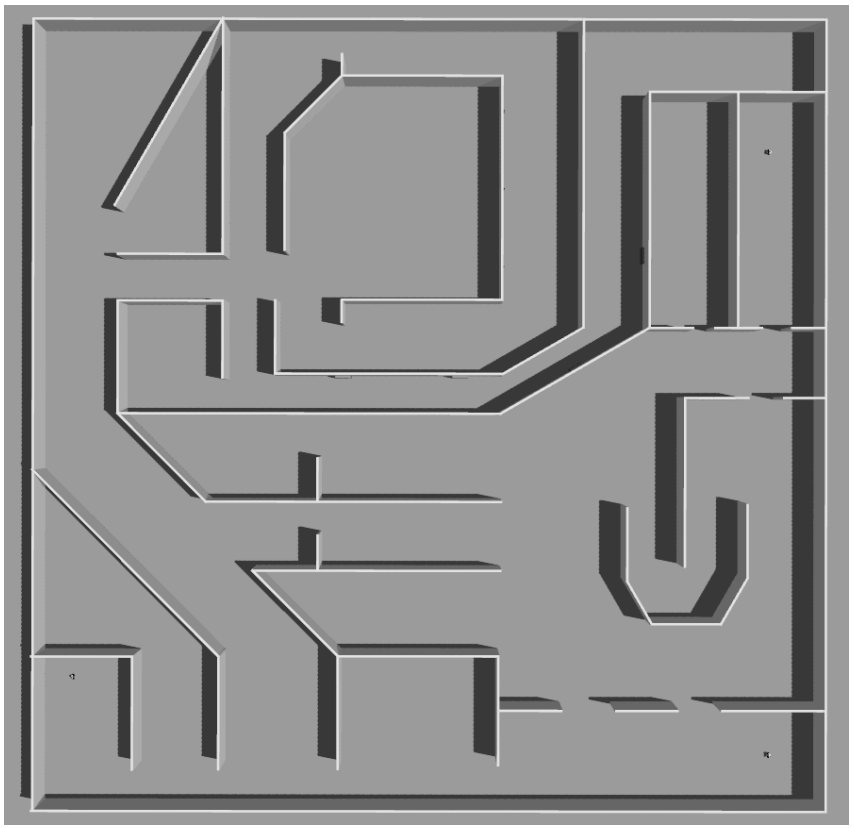


Simulation results

Simulated environments using 1, 3 and 5 robots.



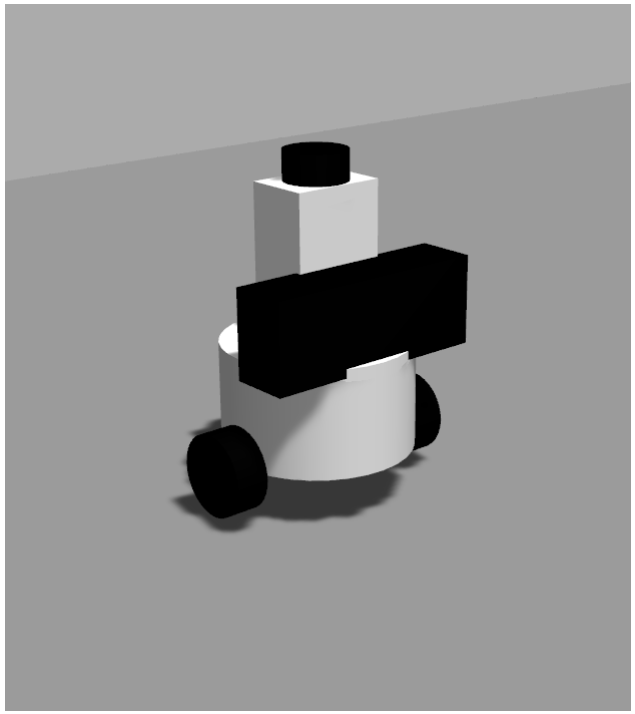
Small environment



Large environment

Simulation results

Simulated robot specifications.



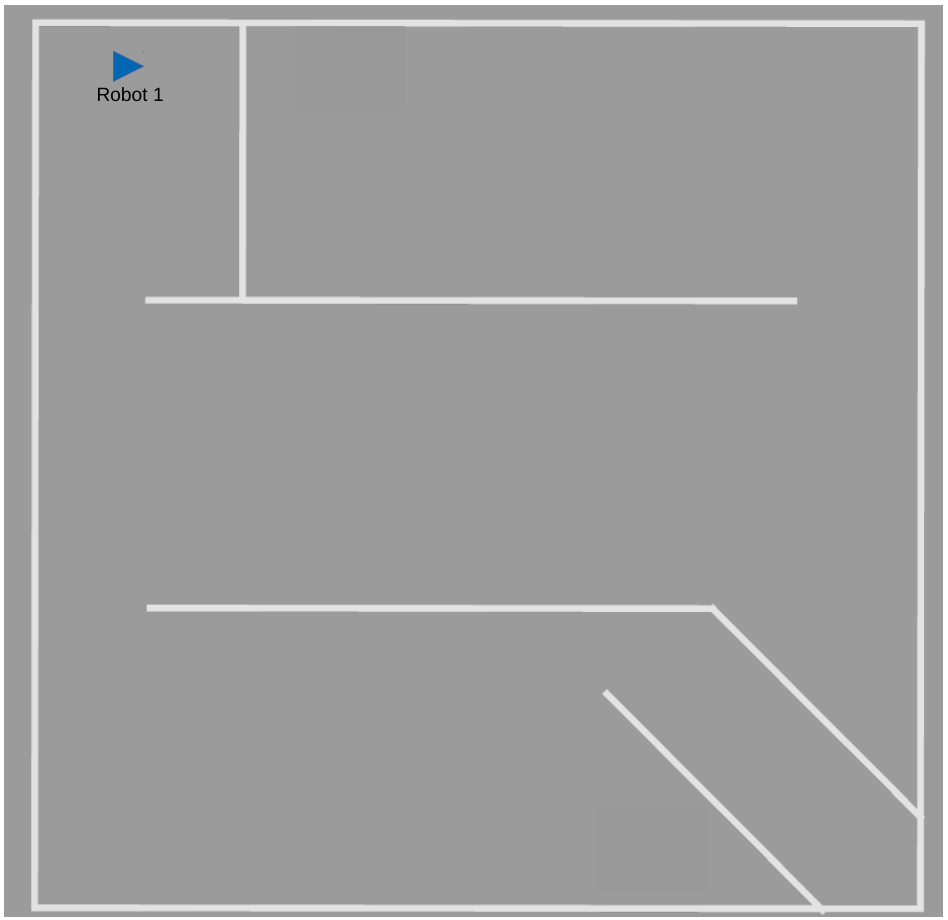
Simulation model

- 2D LIDAR scanner;
- Encoder measurements;
- Inter-robot measurements by ground truth position from Gazebo;

Simulation results

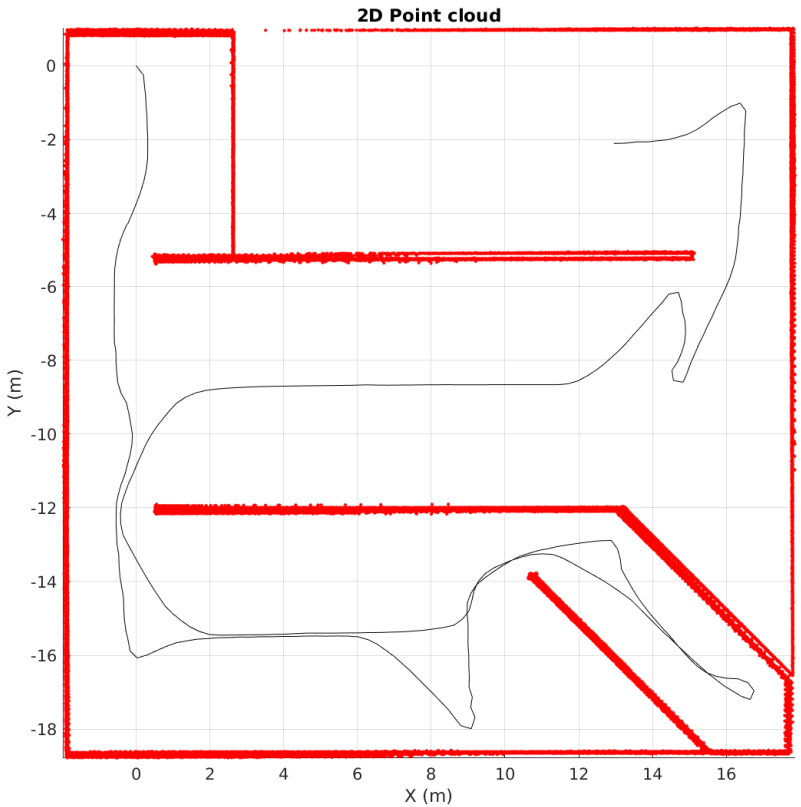
Small environment, 1 robot:

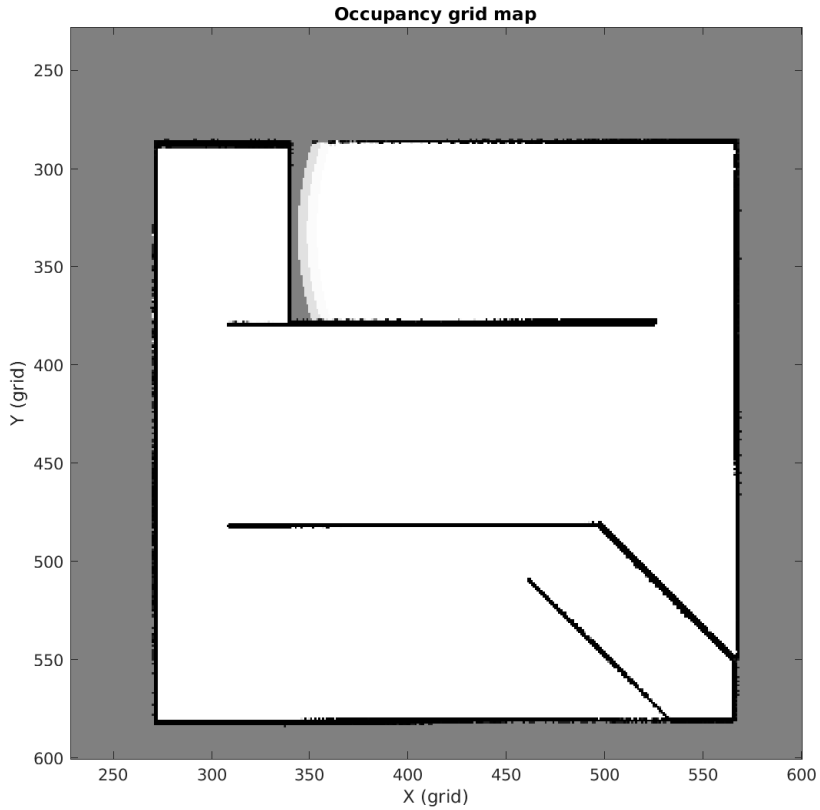
- Total area: 20x20 m^2
- Coverage threshold for each robot: 98% of total area



Simulation results

Small environment, single-robot SLAM.

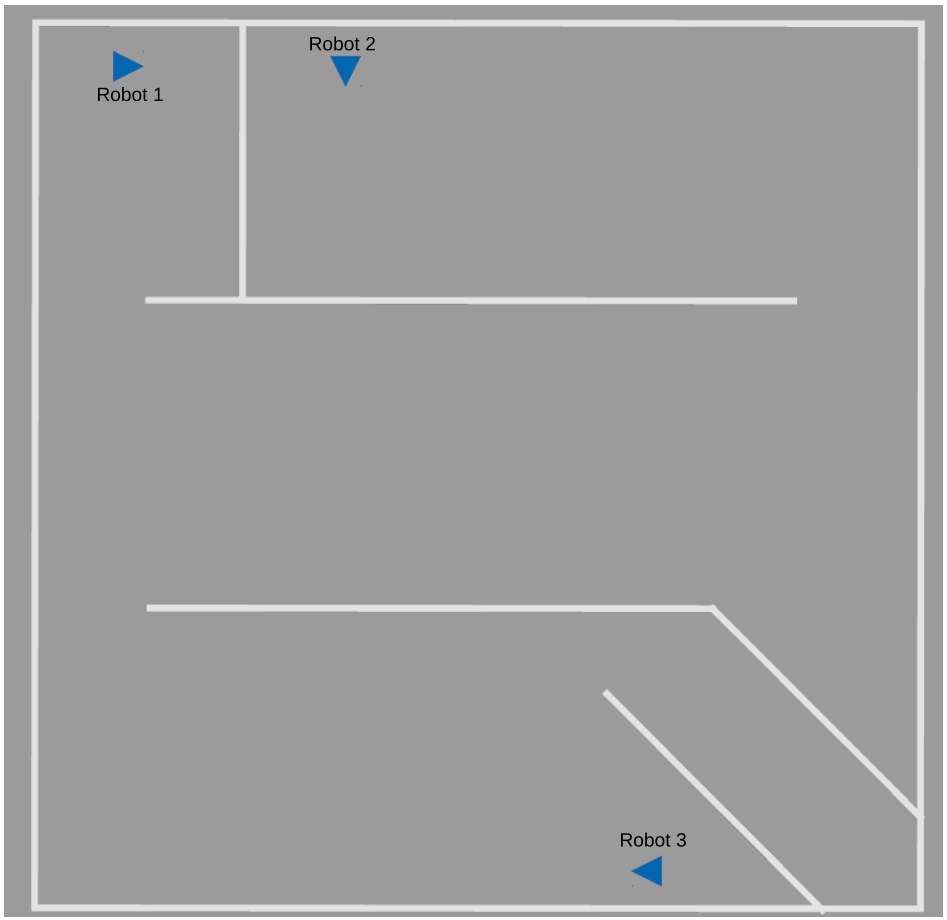




Simulation results

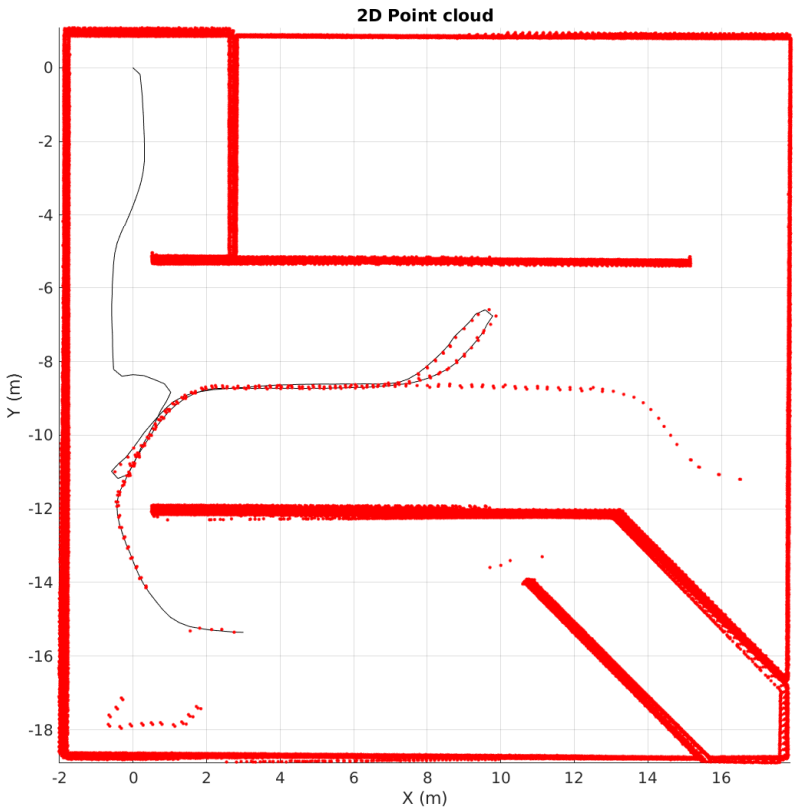
Small environment, 3 robot:

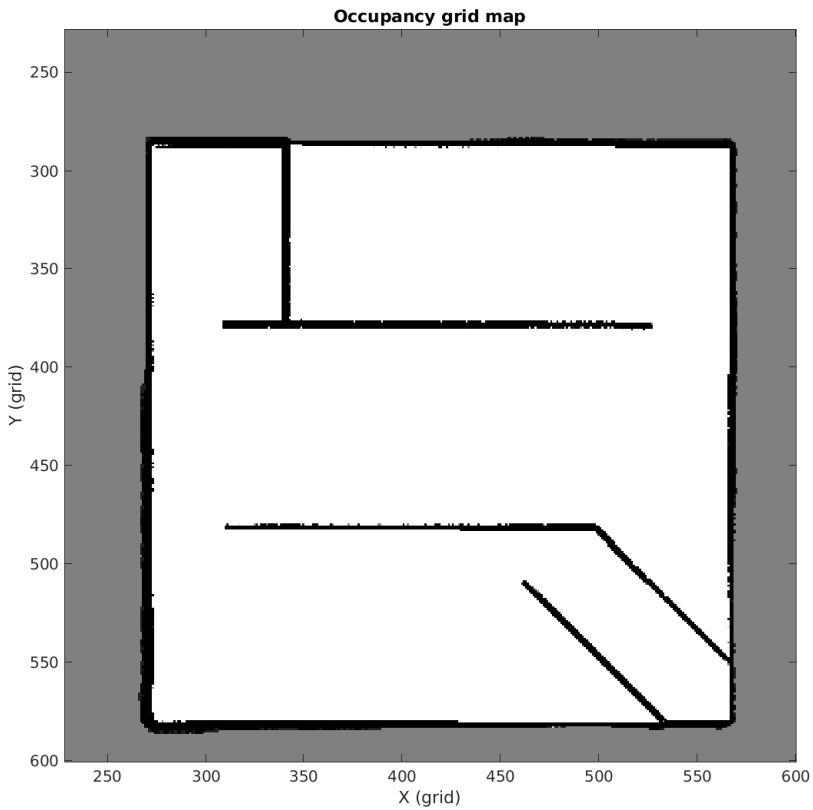
- Total area: 20x20 m^2
- Coverage threshold for each robot: 98% of total area



Simulation results

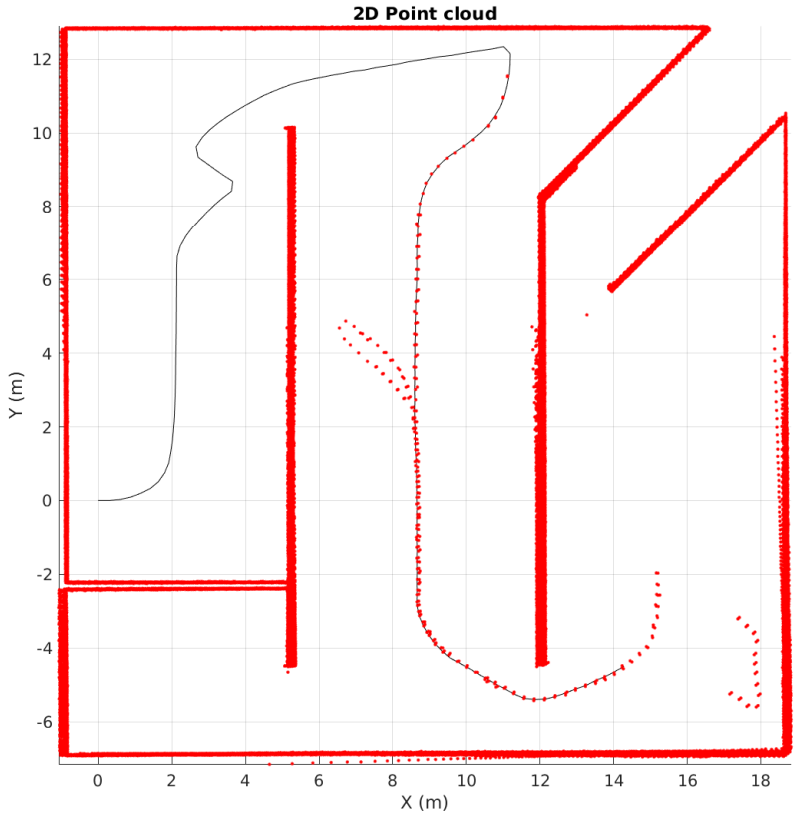
Small environment, multi-robot SLAM, robot 1.

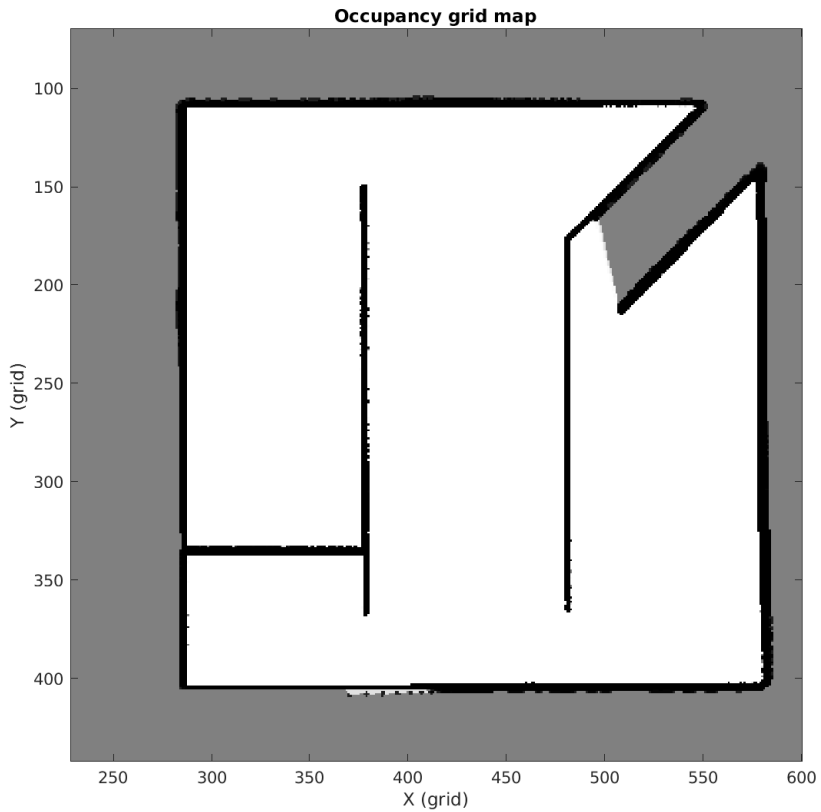




Simulation results

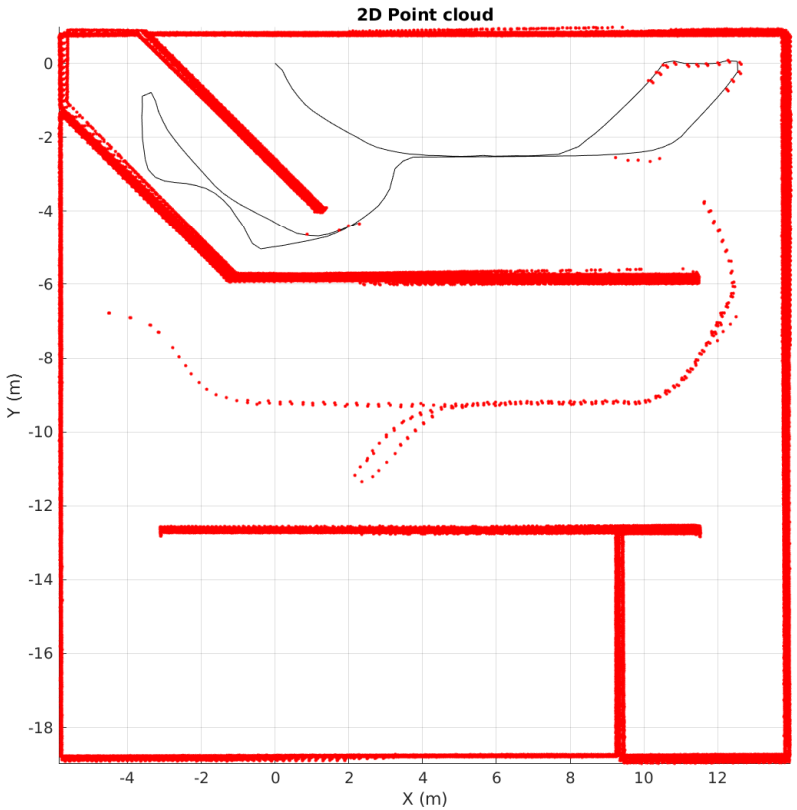
Small environment, multi-robot SLAM, robot 2.

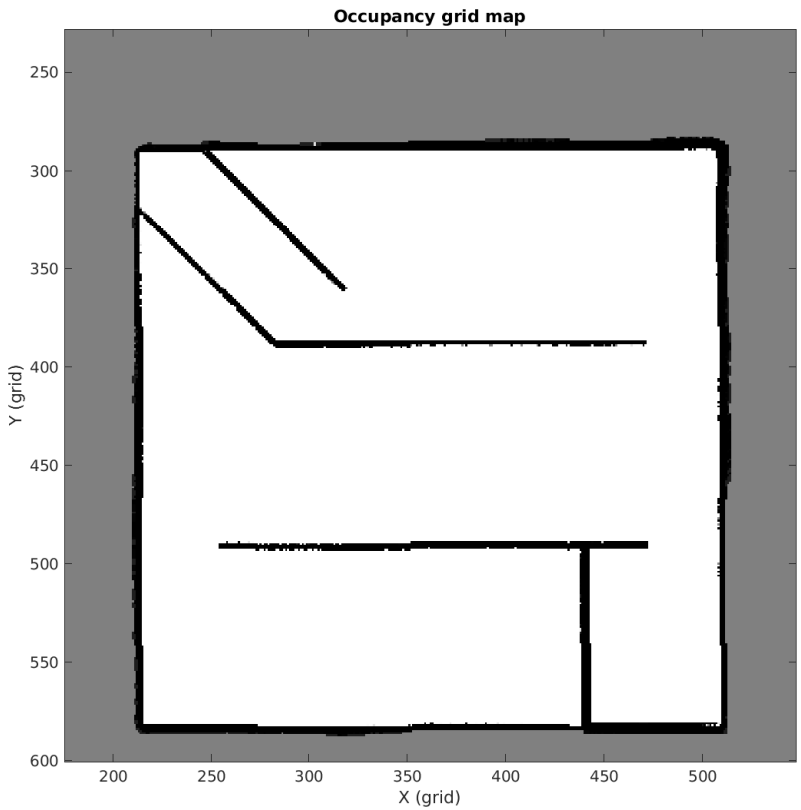




Simulation results

Small environment, multi-robot SLAM, robot 3.





Simulation results

Small environment, 1 robot.

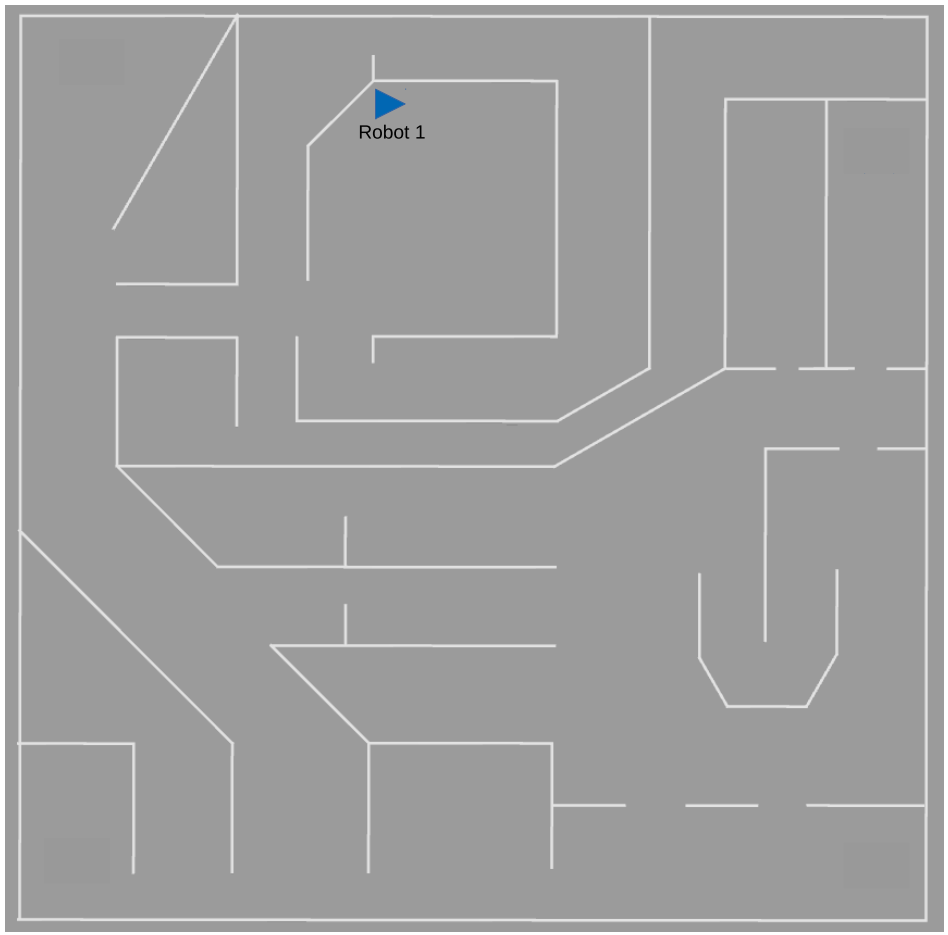
Simulation results

Small environment, 3 robot.

Simulation results

Large environment, 1 robot:

- Total area: 50x50 m^2
- Coverage threshold for each robot: 90% of total area



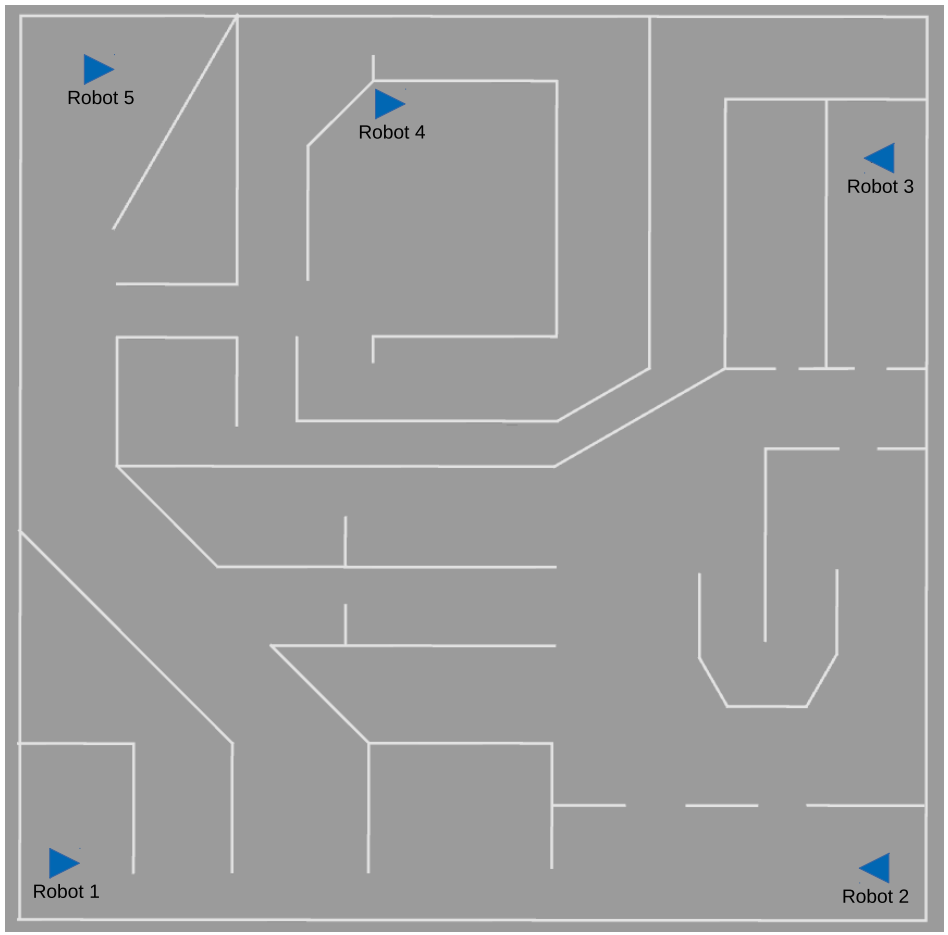
Simulation results

Large environment, single-robot SLAM.

Simulation results

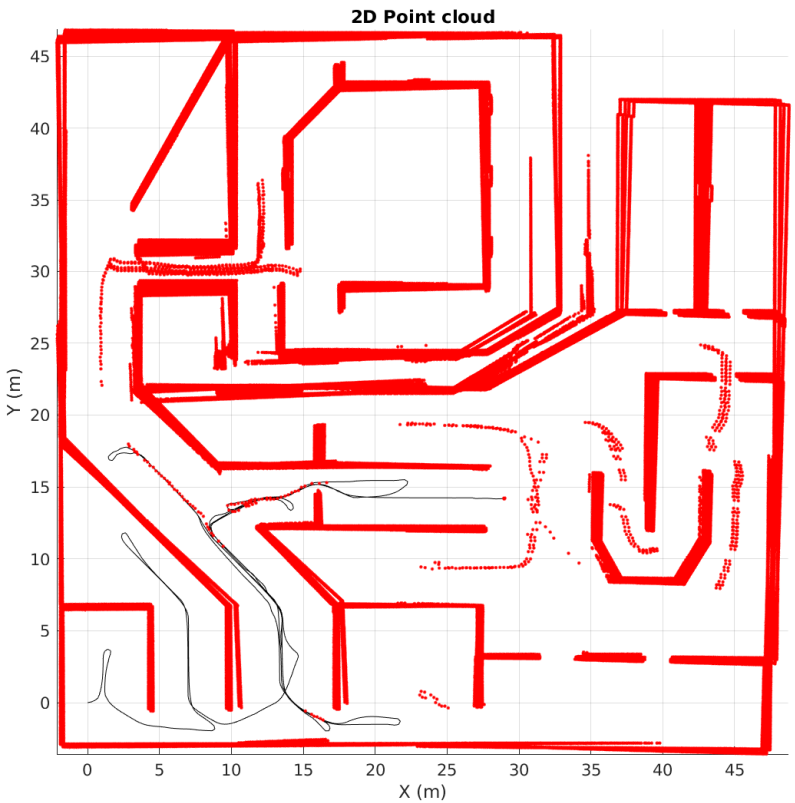
Large environment, 5 robots:

- Total area: 50x50 m^2
- Coverage threshold for each robot: 90% of total area



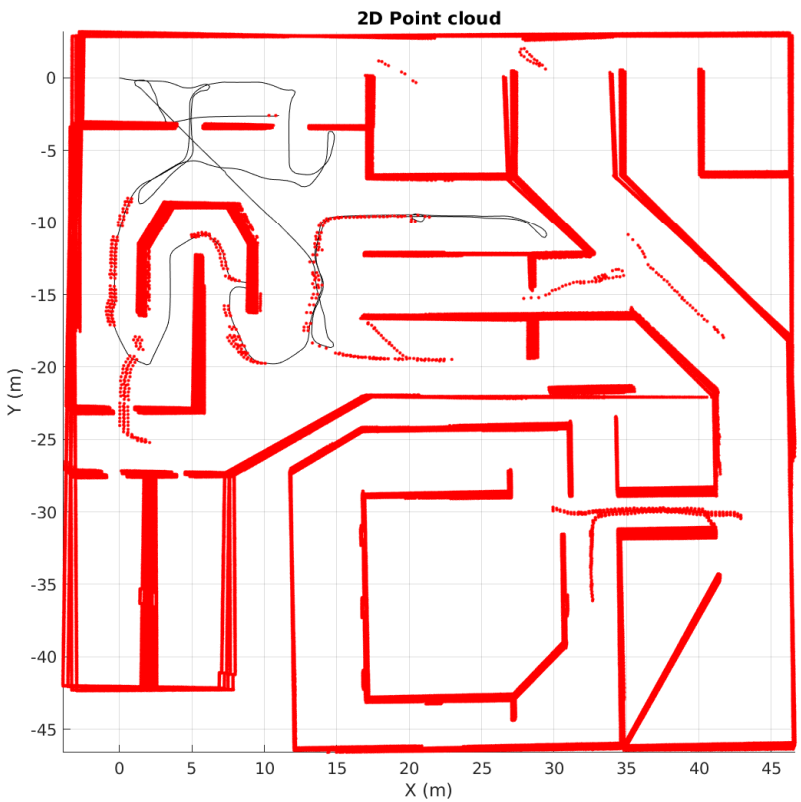
Simulation results

Large environment, 5 robots, robot 1.



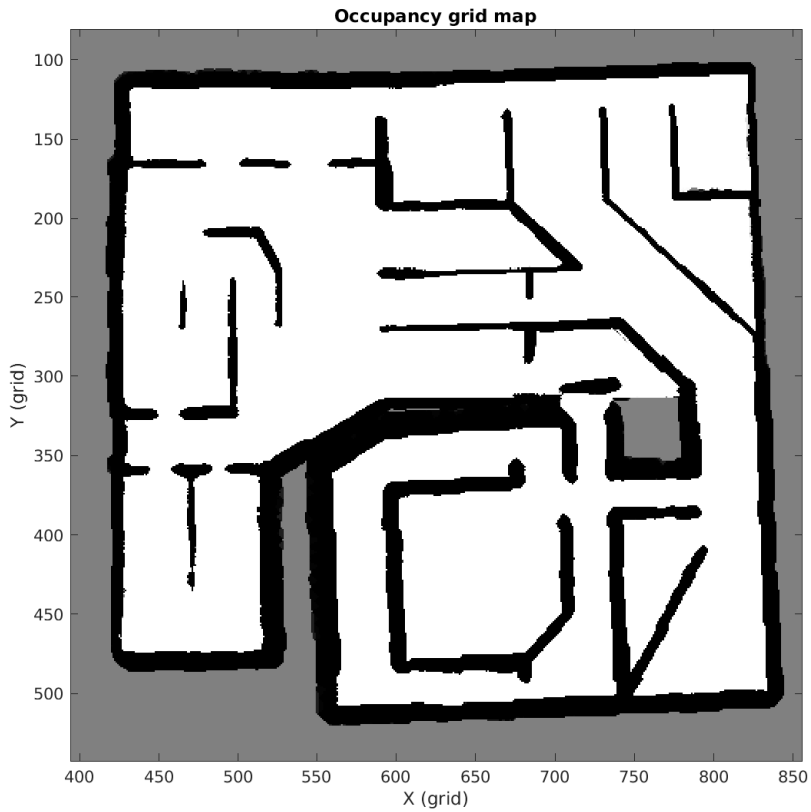
Simulation results

Large environment, 5 robots, robot 2.



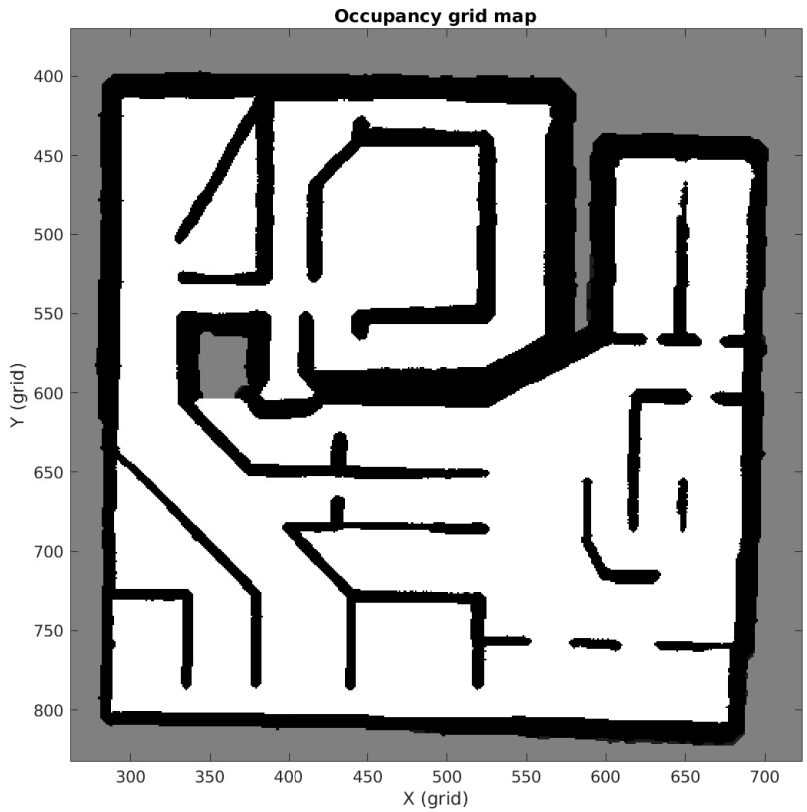
Simulation results

Large environment, 5 robots, robot 3.



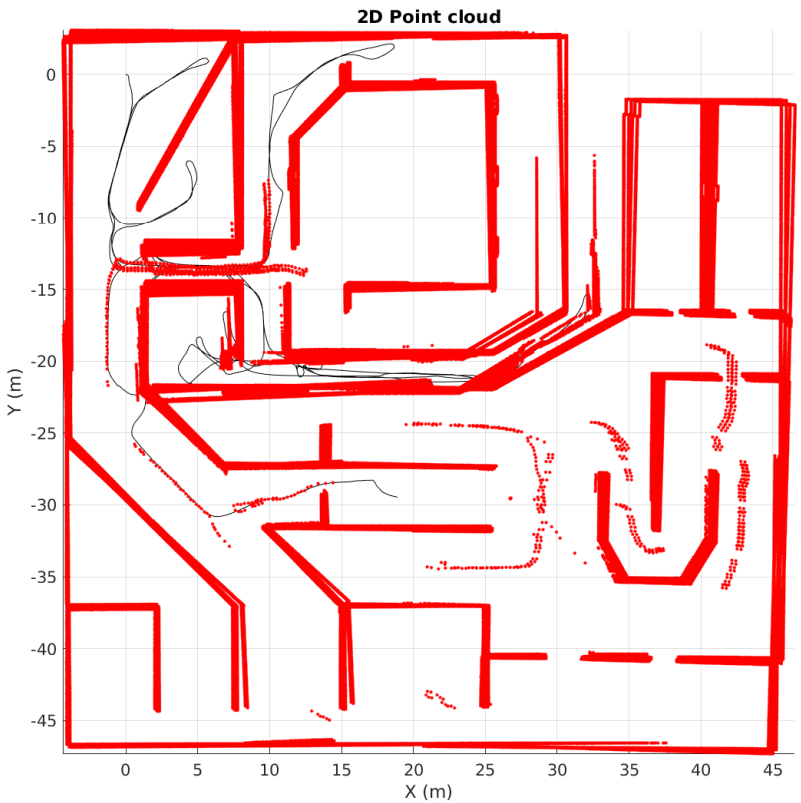
Simulation results

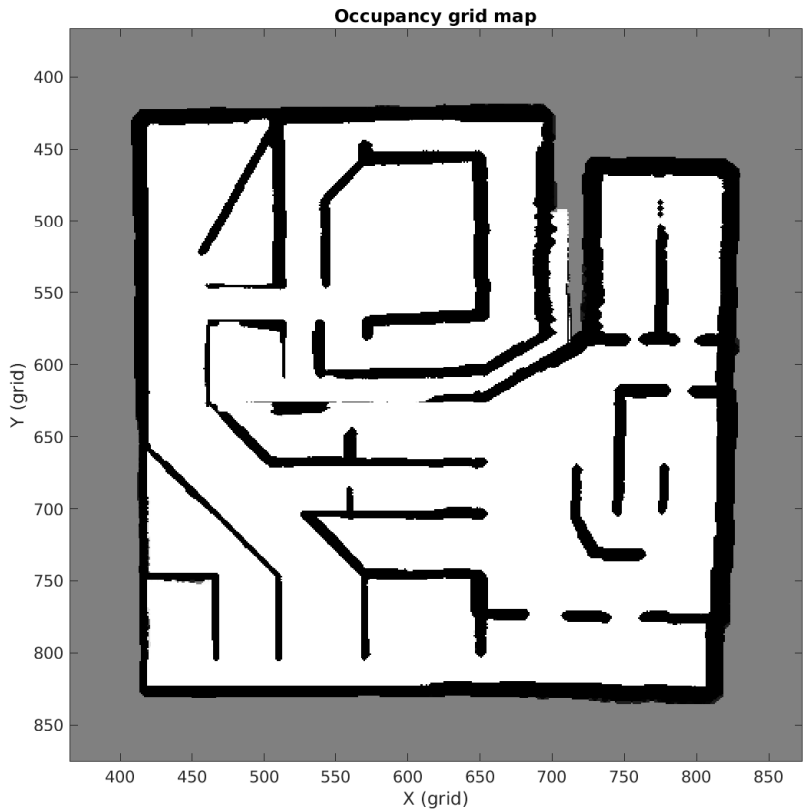
Large environment, 5 robots, robot 4.



Simulation results

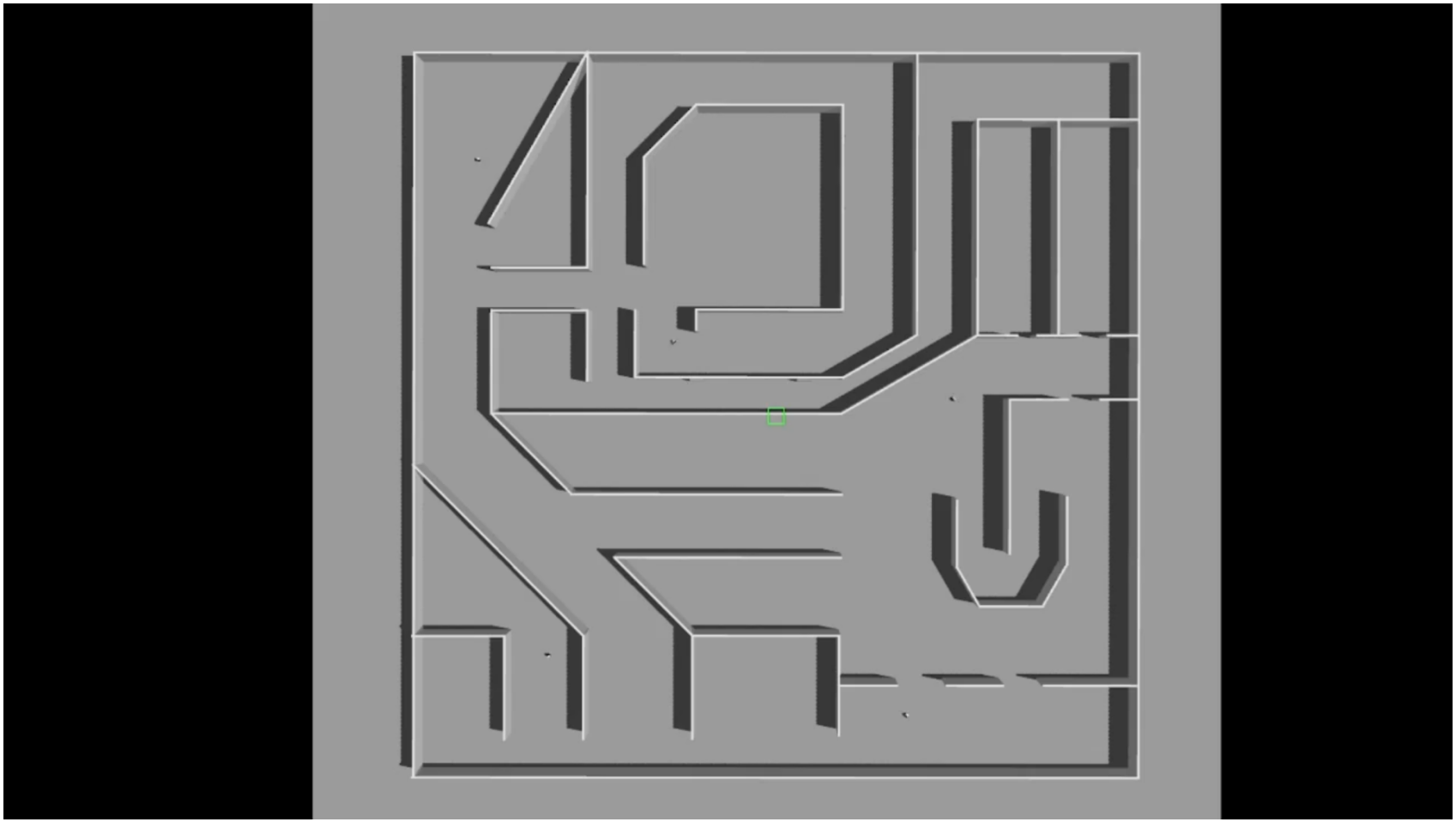
Large environment, 5 robots, robot 5.





Simulation results

Large environment, 5 robots.



Real world results

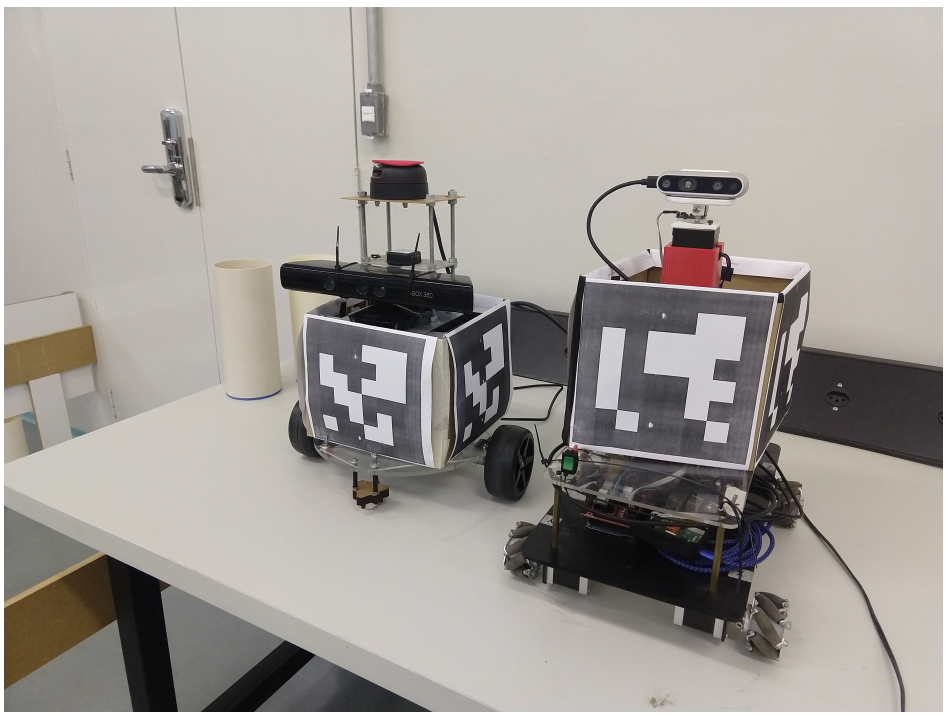
Environments used for the real world experiments.

Lab 1224

LMI corridor

Real world results

Real robots specifications.

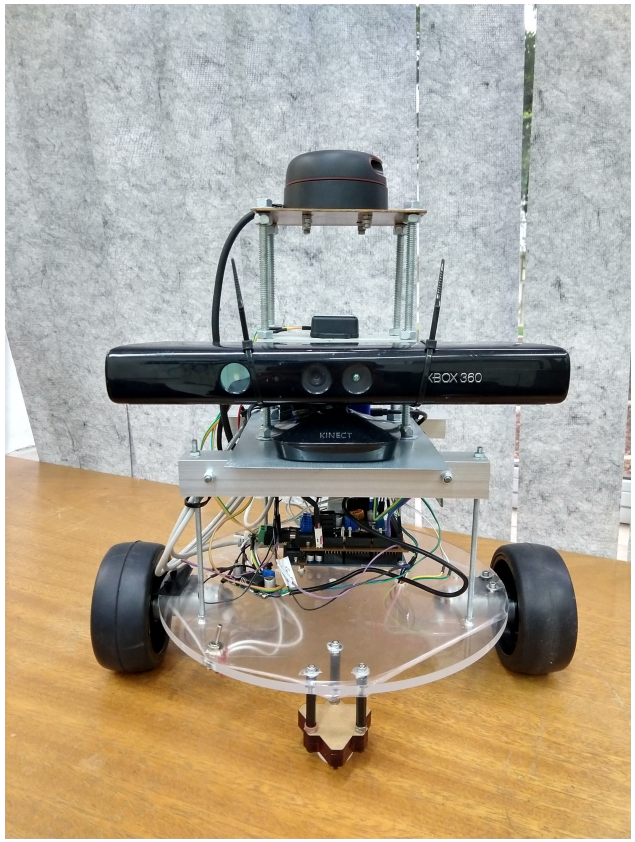


Isis and Omni

- Same measurement height
- Tags printed on all sides

Real world results

Hardware specifications of robots.

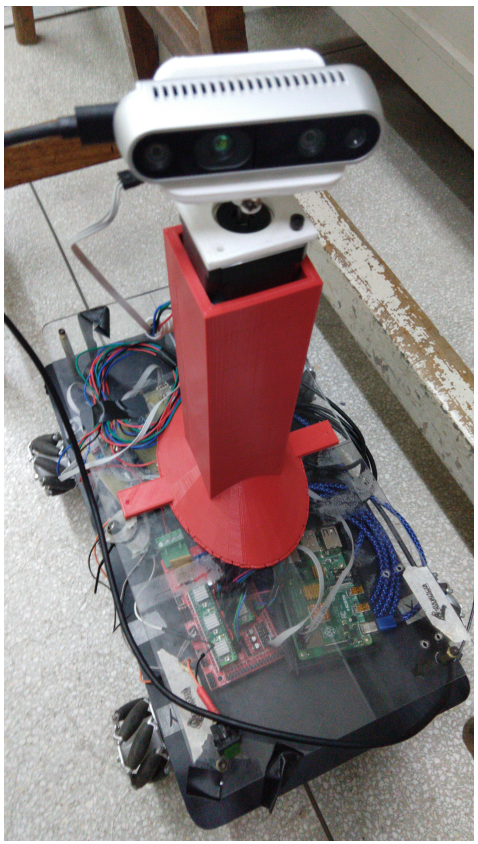


Isis robot

- Processing boards Raspberry Pi 3B+ and Arduino Mega 2560
- 2D LIDAR SLAMTEC RPLIDAR A2M8
- 2x DC motors, 6V, 100 RPM
- Wheel encoders
- RGB-D camera Kinect-for-Xbox
- IMU WiT BWT901CL
- Inter-robot measurements by tag detection

Real world results

Hardware specifications of robots.

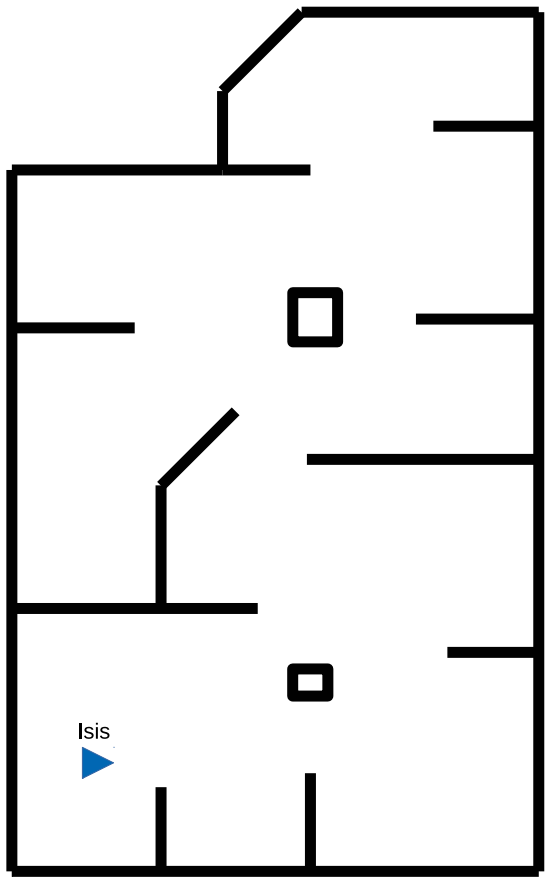


Omni robot

- Processing boards Raspberry Pi 4B and Arduino Mega 2560
- 4x Stepper motors, 12V, NEMA 17
- RGB-D camera Intel Realsense D435i
- IMU Sparkfun LSM9DS1
- Inter-robot measurements by tag detection

Real world results

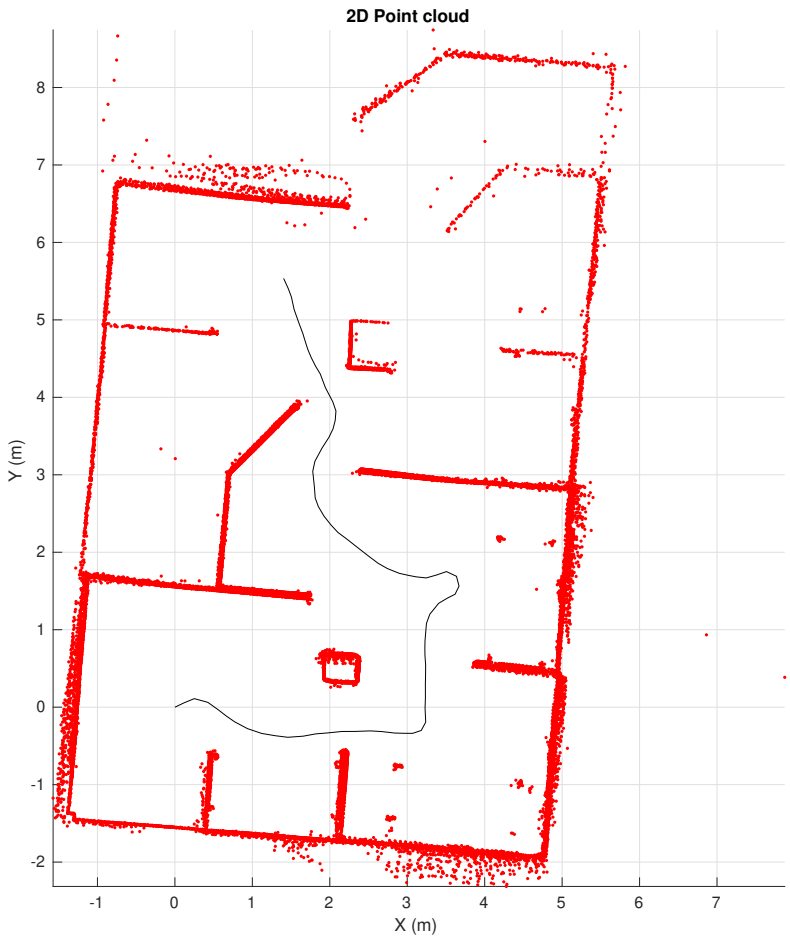
Lab 1224, 1 robot:

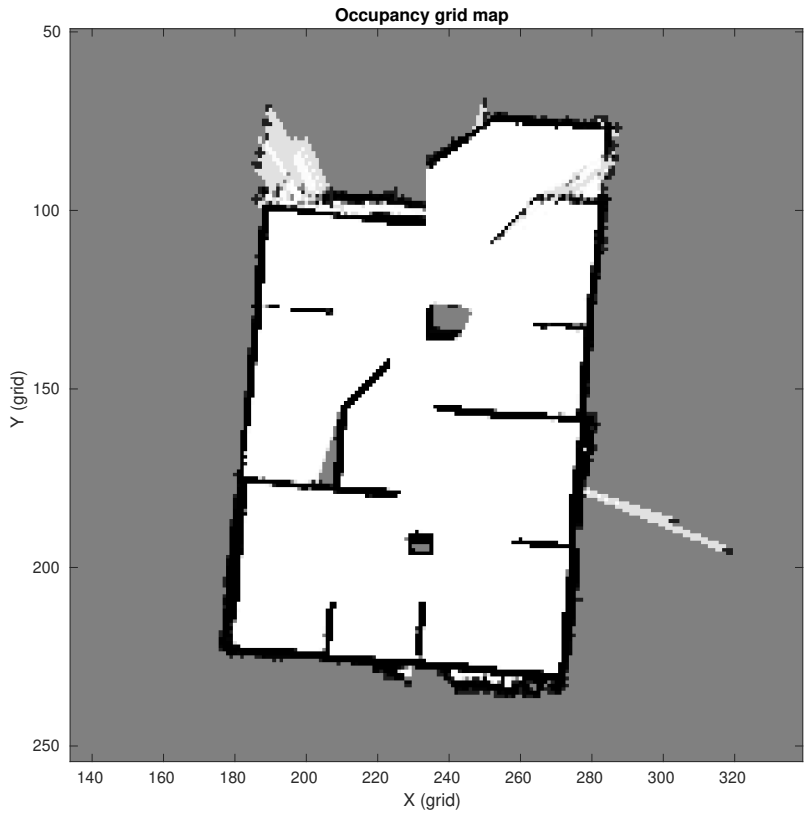


- Using only Isis
- Total area: $\approx 53.86\ m^2$
- Coverage threshold: 100% of total area

Real world results

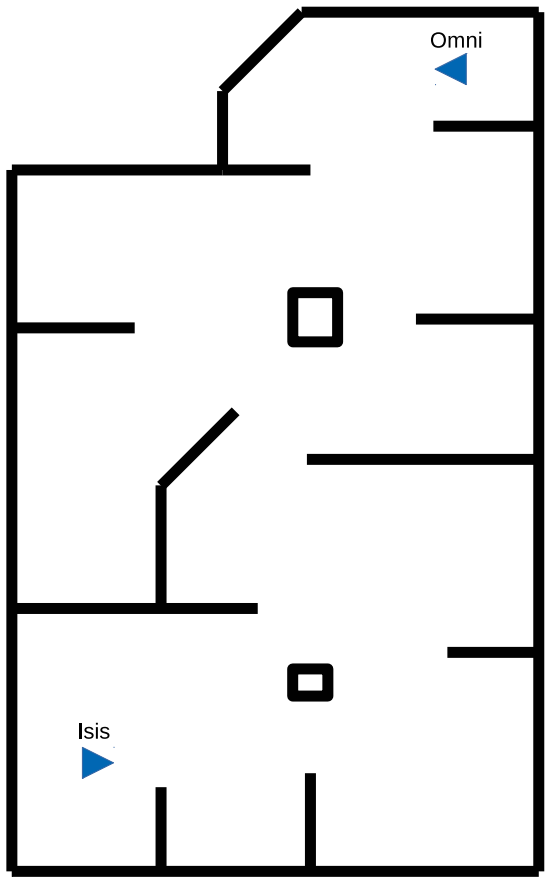
Lab 1224, Isis robot, single-robot SLAM.





Real world results

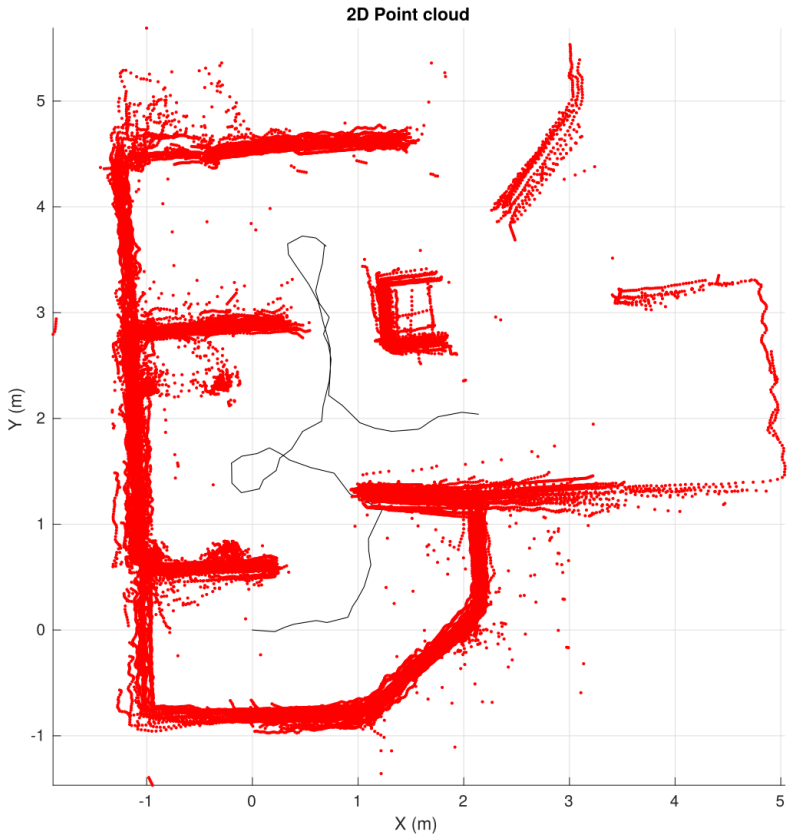
Lab 1224, 2 robots:



- Using Isis and Omni
- Total area: $\approx 53.86\ m^2$
- Coverage threshold: 100% of total area

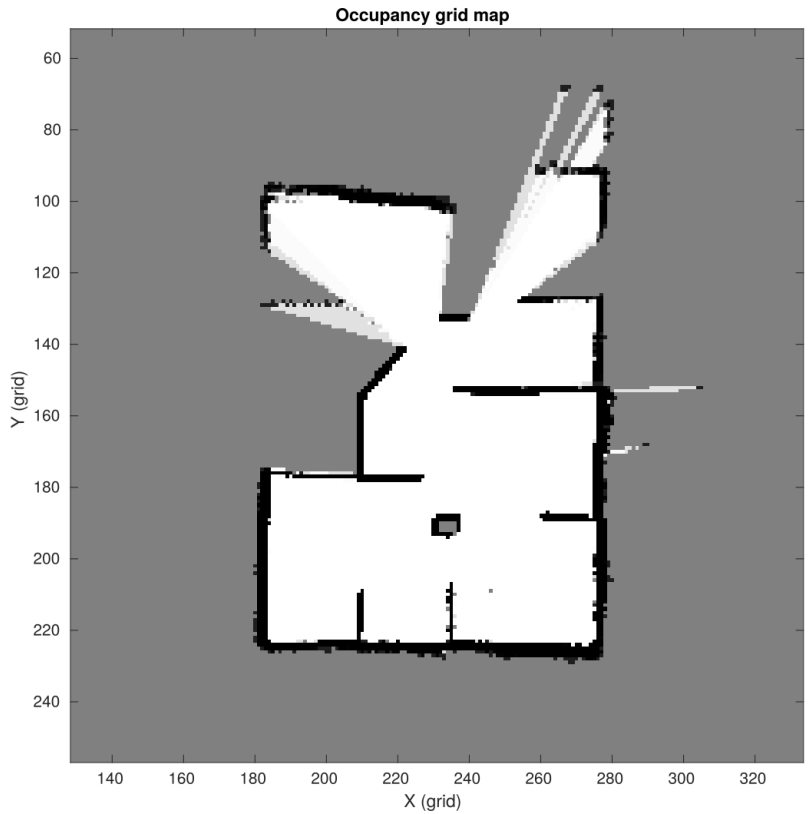
Real world results

Lab 1224, Omni robot, multi-robot SLAM, map before merge.



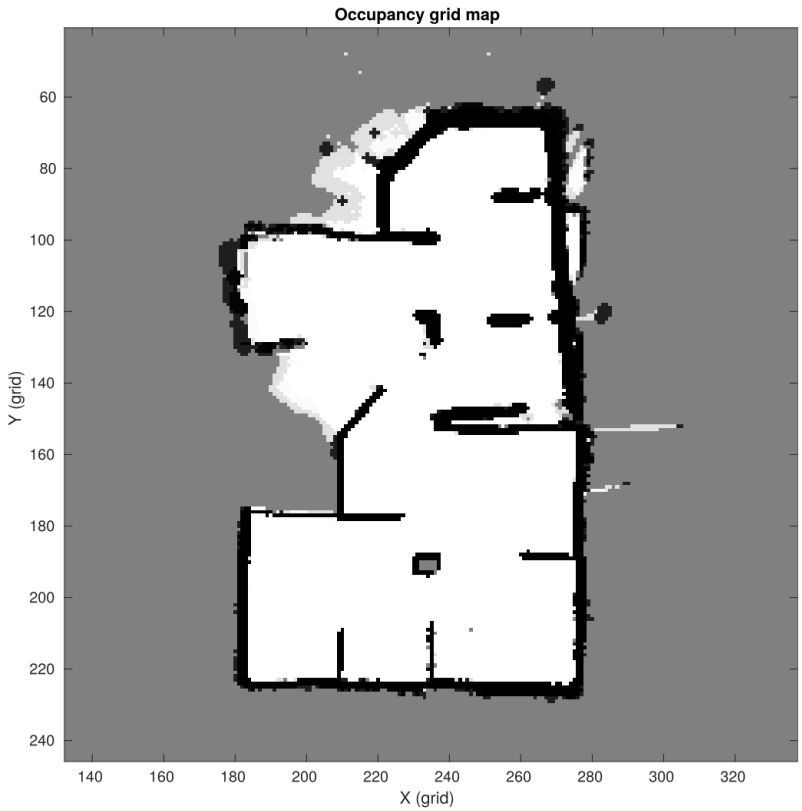
Real world results

Lab 1224, Isis robot, multi-robot SLAM, map before merge.



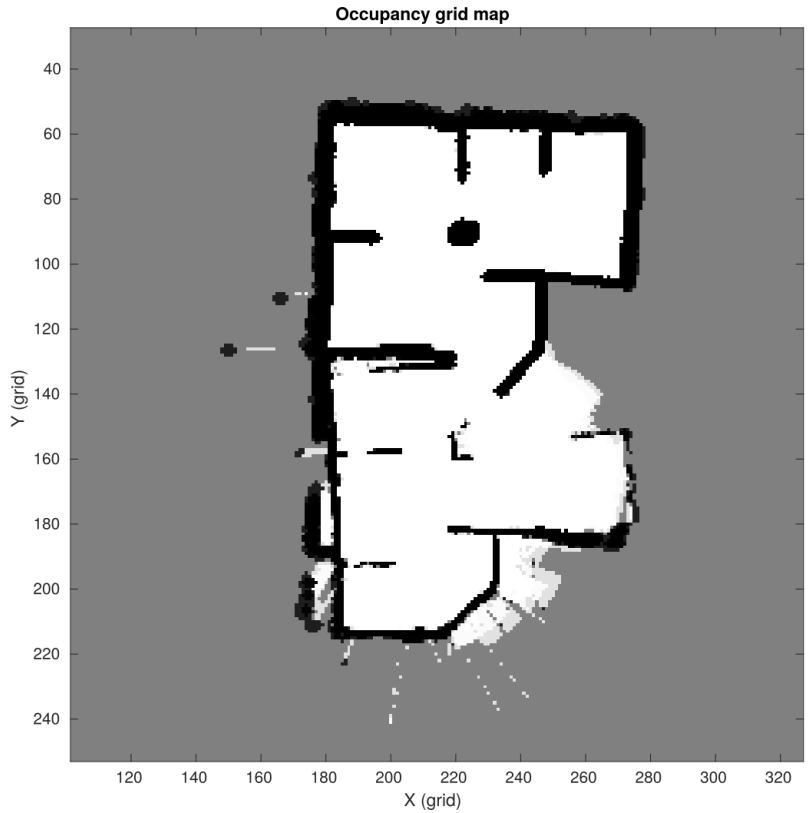
Real world results

Lab 1224, Isis robot, multi-robot SLAM, after merge.



Real world results

Lab 1224, Omni robot, multi-robot SLAM, after merge.



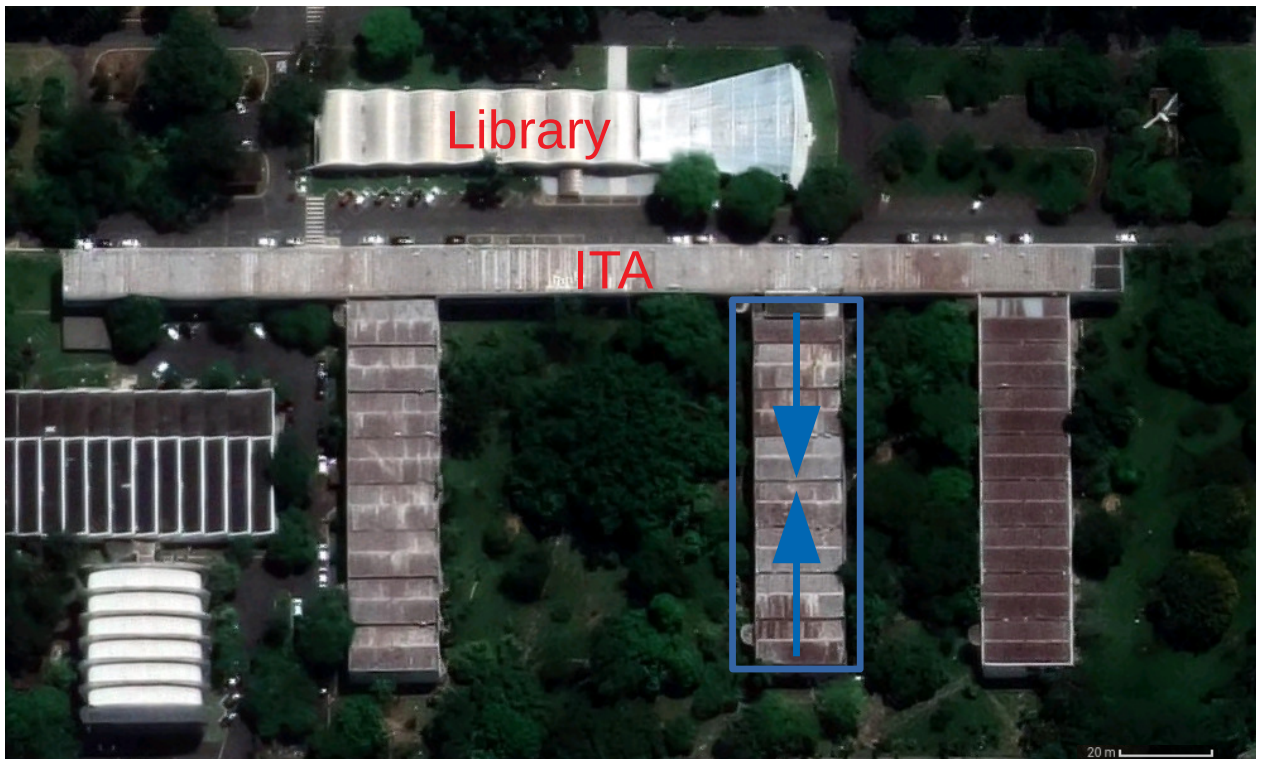
Real world results

Lab 1224, instant of map merge.

Real world results

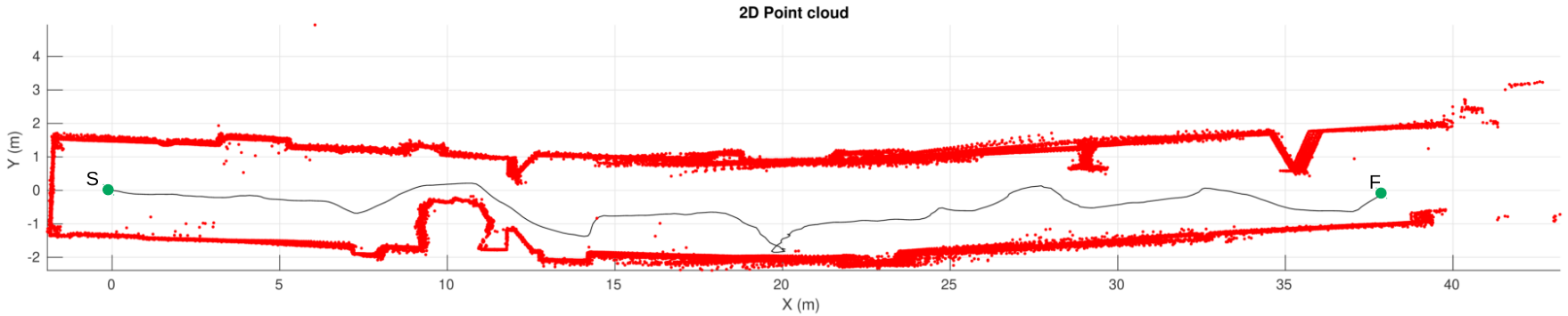
LMI corridor, 2 robots:

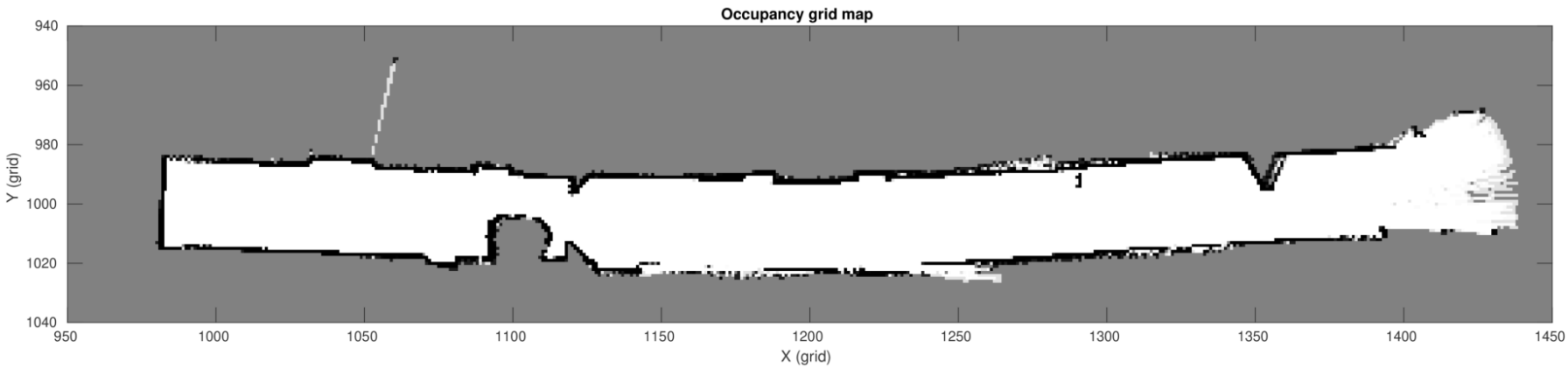
- Using only Isis
- Map merge is processed offline
- Rendezvous at the corridor middle point



Real world results

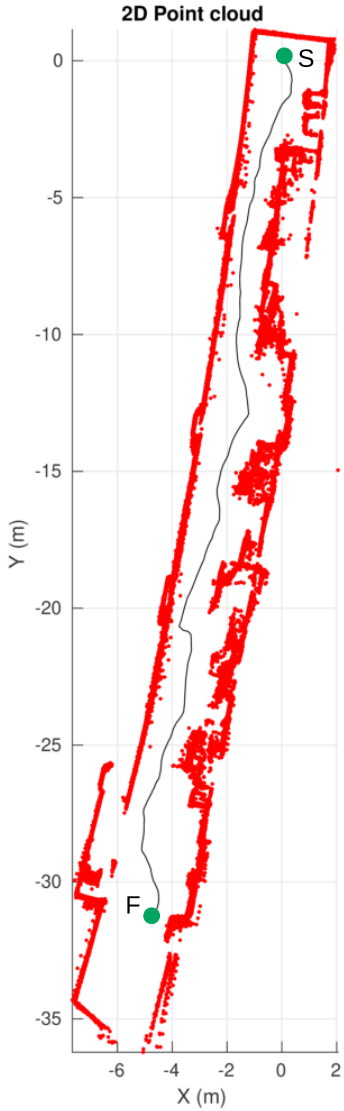
LMI corridor, Isis robot, first half.

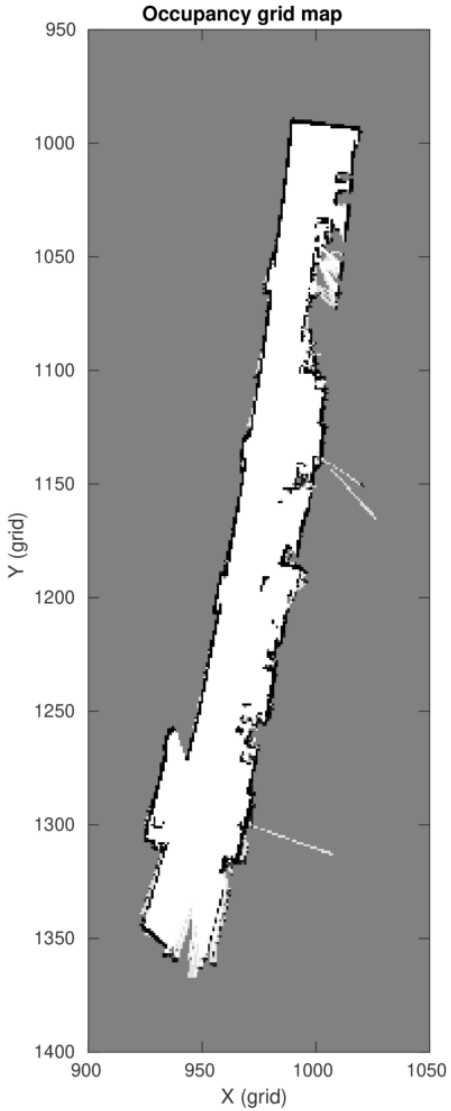




Real world results

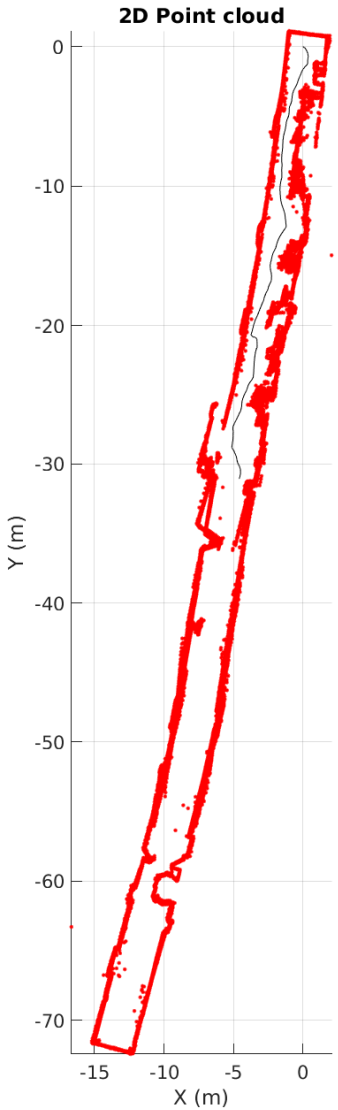
LMI corridor, Isis robot, second half.

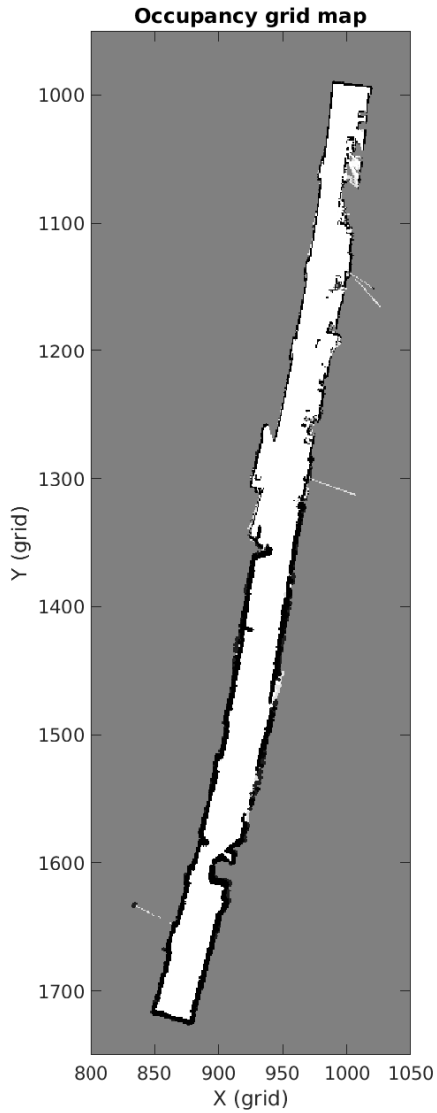




Real world results

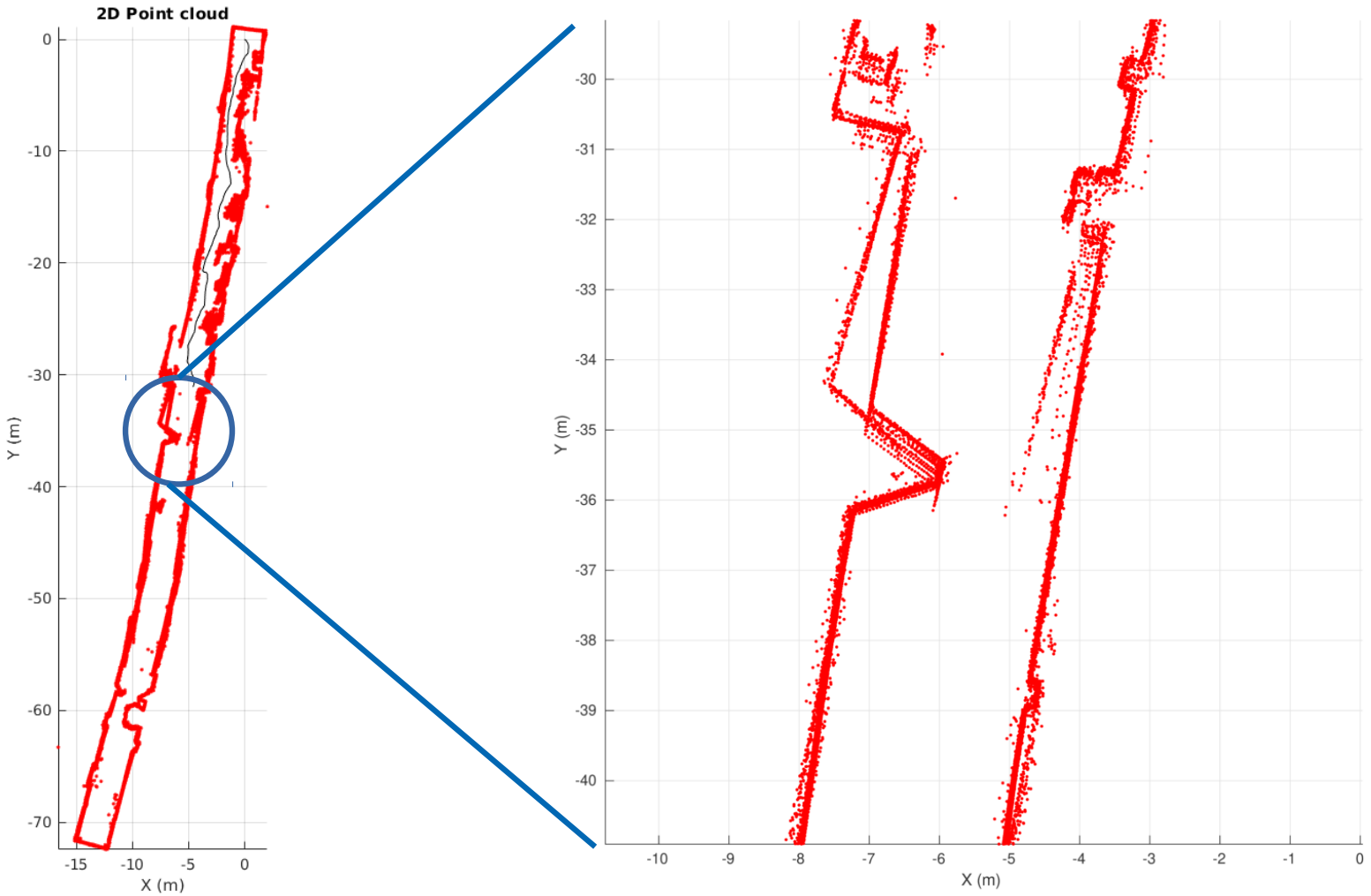
LMI corridor, Isis robot, merged maps.





Real world results

LMI corridor, Isis robot, zoom in merged point cloud.

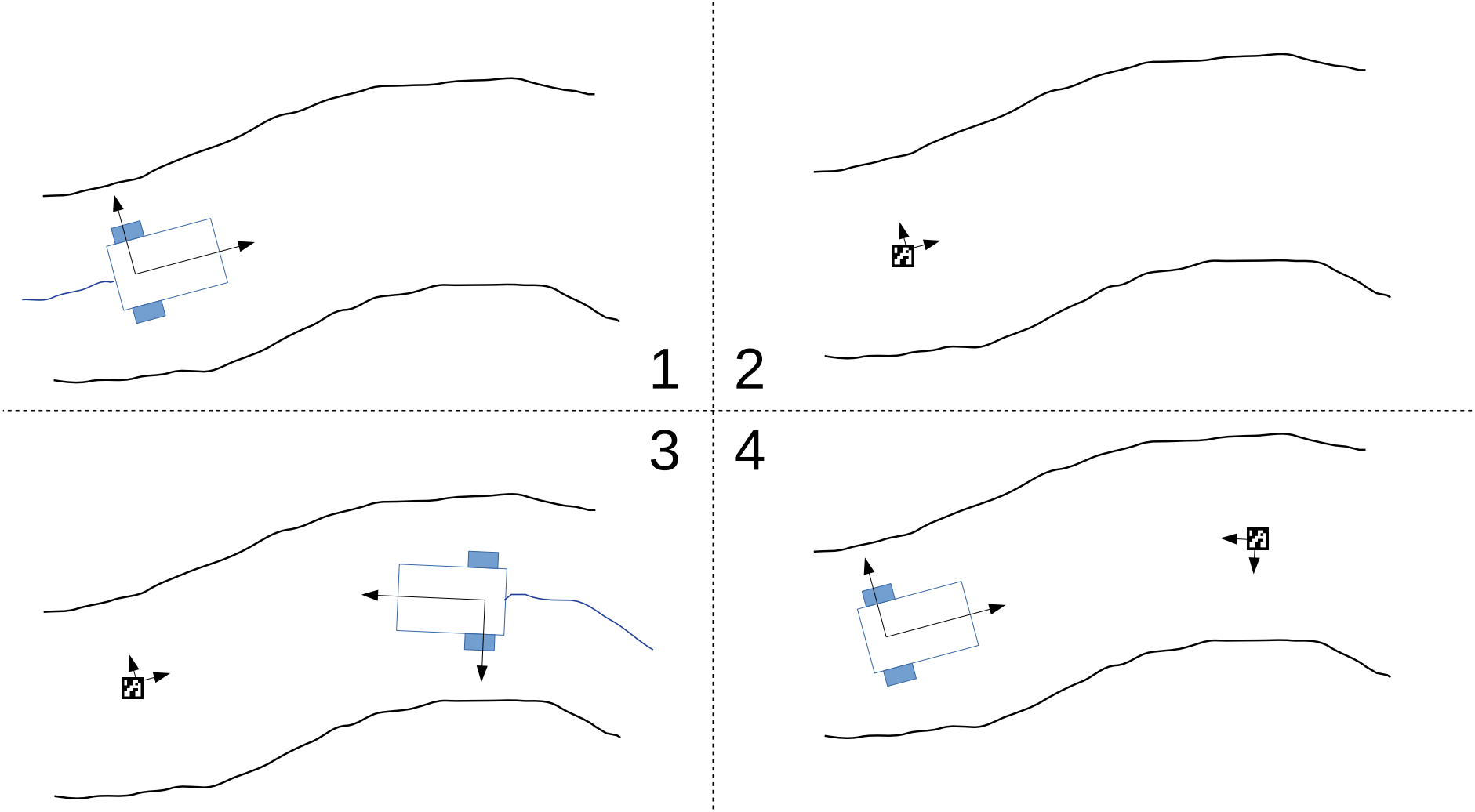


Real world results

LMI corridor, second half.

Real world results

Offline map merging using only one robot.



Conclusion

- Using the proposed solution, multiple robots to perform active SLAM directly implies faster task completion by all robots, as foreseen by the literature.
- The use of a simplified coordination rules was shown to be sufficient to solve the problem of avoidance of repeated target assignment for the scope of the experiments. With increased number of robots, this strategy may not hold as expected.
- Real world experiment results showed that the solution provided satisfactory results.

Future works

1. Validation of the proposed solution in outdoor environments, as a mean to generalize the approach. In this context, the objective will not be complete coverage, but it will be a search and rescue application.

2. The extension for 3D estimation for both maps and poses, RGB-D cameras can be used as a sensor for this kind of estimation.

3. Make the architecture a decentralized one, where the robots do not depend on a central computer.

4. Validation with a larger team of robots using a more sophisticated set of rules for coordination.

Acknowledgments