0/1 Questions Answered

PA4 - Auction-based task allocation

Q1 The problem

10 Points

Goal

The purpose of this assignment is to introduce you to auction-based mechanisms and learn how to implement it.

Instructions

Please read carefully the following tasks and write the program(s) accordingly. *You should do this assignment individually*. Feel free to reach out on Slack in the #help-assignments channel for questions.

Please look also at the general guidelines for writing code:

https://canvas.dartmouth.edu/courses/46138/pages/notes-on-coding-for-the-assignments

Getting ready

Please revise the material we covered so far, in particular the latest about ROS custom messages, services, actions, and tf. You can find the files used as well as additional readings on the description of each day in the calendar. The mechanisms to allow communication between robots were set for PA3, thus you can start from that code to implement an auction-based mechanism.

It is assumed that the ROS environment is already set up with both Gazebo and/or `stage_ros`.

The task

The robots should coordinate to cover specific locations in a common reference frame. To simplify the problem, you can assume that only one robot is the auctioneer and that robots bid on a single location independently. The auctioneer then assigns a task to robots following the sequential auction mechanism, until each task is allocated to a robot. The test can be done in an empty world with 3 robots and the locations are randomly generated over a space of 10m x 10m.

Here a guideline that you can follow to achieve this behavior:
1. write a node that publishes random waypoints that the robots should cover. There should be a check to ensure that waypoints are not too close to each other. These waypoints will be the items that the auctioneer will ask bidders to bid for during the auction time.
2. write a node for the auctioneer that sends waypoints for the other robots to evaluate and according to the bids, return the allocations to each robot following the sequential auction mechanism.
3. write a node for the bidders that return the values associated to the waypoints.

Report

Please write:

1. a report in markdown in a file called `report.md` to discuss your implementation. Then use pandoc or something similar to generate a pdf. Include both files in your submission, as well as all figures (as pdf). The report should contain the following information:

   a. Header with the code of the class, the term, and year, the assignment number, and your name.
   b. A short typewritten report (at least one full page of complete sentences) describing your results. The report should contain three sections:
   (a) Method description: How does your program work? What design decisions did you make?
   (b) Evaluation: Does your program actually work? How well? If it doesn't work, can you tell why not?

2. a README file in markdown with the instructions on how to execute the program.

Video

Please record also a short video showing the robots coordination in action.

Q1.1 Grading rubric and submission
10 Points

Grading

Your programs will be evaluated based on both their functionality and their coding style. In the notes for writing programs, you can find an informal style guide to help give you an idea of what is expected together with the coding style that you should follow. In particular, the following aspects are considered:

Performance (7):

- Correct generation of random waypoints.
- Correct auctioneer mechanism.
- Correct bidder mechanism.

Style (1.5):

- No duplication of executable code?
- No magic numbers?
- Names match functionality?
- Adequate comments?
- Comments match code?
- Consistent formatting?

Documentation (1.5):

- Report is complete and clear.
- Required sections exist under readily identifiable headings.
- Free of typos and grammatical errors.
- Video included

Submission

Please submit a single zip file containing the full ROS package, README, report, and video. You are welcome to write any comment about the assignment/submission in the following text box.

Enter your answer here

Please select file(s) Select file(s)

Save Answer

Save All Answers Submit & View Submission >