

Mission planner test instructions

Before starting

- Make sure that everything is running (instructions for this on GitHub: <https://github.com/joakimgrutle/Multi-robot-mission-planner>)
- Refreshing the webpage will put everything back to default, so if anything goes wrong, or you just want to reset, simply refresh the page
- DISCLAIMER: this is not a finished product, it is more of a proof of concept, the system contains some bugs, poor error handling, limited amount of robots and actions, the user interface design is suboptimal, and algorithms (and such) that are used are not optimised for the system and all of this is open for improvements. However the functionality of the system should be in place to see that the concept would work (in theory), and this is what you are testing by going through these instructions.

Connecting to Webots simulation stream

- Connect to the simulation stream by scrolling down to 'Webots streaming viewer' and click on connect (should by default be set to 'localhost' and '1234')
 - A simulation player should then show up, and be connected to the local instance of Webots
 - Now everything should be ready for running missions in the browser
 - NB! The stream from the simulation has a tendency to start lagging after being used for a while, if this happens, simply press disconnect and then connect again. You could also refresh the page, but this is not recommended for later because this will remove all changes to missions

Running a mission

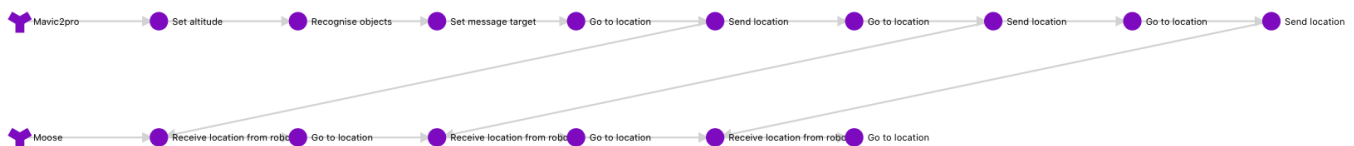
- The simulation is interactive, meaning that missions can be sent and altered during execution. Running a mission is a simple two step process:
 1. Start the simulation, this is done by pressing play in the simulation window at the bottom of the screen:



2. Press the 'Send mission' button to send a mission, and the robots in the simulation should start moving (if you successfully started the simulation).
 - Pressing 'send mission' before the simulation is running will not do anything until you press play on the simulation to start it
 - The order of these steps does not matter, but the server will stop trying to send the mission into the simulation after trying for one minute.
- To reset the simulation, simply press the '|<<' button (see image above), this makes everything in the simulation go back to how it was in the start
- The mission planner comes with some predefined missions, you can try to run them by selecting a mission from the top, run it as described above, and reset when you are happy
- The missions will pause automatically every minute in order to not use too much processing power, just press play to continue a mission if it pauses

The mission timeline

- Pressing the 'Show mission timeline'-button will show the timeline of when each robot will execute each simple action, and the order of execution. This should be useful to see how a mission will execute. The mission timeline for the quarantine mission should look something like this:



- A task cannot be executed before tasks who points to it has finished executing
- You can drag and zoom on the timeline if parts of it is outside of the view

Modifying missions

- You can first try to change the predefined missions by simply dragging and dropping the 'tasks' or the 'simpleactions', or even try removing them by pressing the red X
 - Dragging around the tasks will have little effect unless there are more than one task for one robot (this would then change the order of which that robot does its tasks)
 - Dragging around simpleactions will change the order of how a robot will execute, and this might change the mission or possibly break it, depending on how you position the simpleactions
 - You can mess around with the missions as much as you want and try to run the missions if you desire (by following the instructions above)
- The mission 'Testmission' is a mission made for use in this test.
 - This mission has no assigned robots to its 'tasks', so you can try to press the 'automatic task allocation' button. This will run an algorithm that automatically proposes which robot should do which task in the test mission
 - After pressing the automatic task allocation, then the mavic2pro drone should be assigned to both tasks. This is because the mavic2pro is best suited to perform these simple tasks, and is therefore assigned both.
 - If this is not desirable, then you can manually change which robot is assigned to which task by pressing the dropdown menu next to the task name. The dropdown will show all robot that can execute that task
 - You can try to run the missions after robots have been assigned to tasks
- You can now try to add simpleactions to the mission. You can try adding a 'go forward' simpleaction to the 'test mission'.
 - Make sure 'test mission' is selected at the top, then select task: 'test: forward turn'.
 - To add 'go forward', you can either drag and drop it from the list to the right (under robot simpleactions), or you can search for a 'go forward' using the 'search for simpleactions' button.

- This would add a 'go forward' simple action to the task, but with no argument (empty box next to the name). 'Go forward' takes an argument in seconds. So adding the number 5 to the box next to the name will make that robot go forward for 5 seconds. (hover over input field to see tooltip)
- After you have done this, it should look something like the image below

Missions

Quarantine delivery Scout location and deliver item Scout area and deliver item **Testmission**

Mission name Add new mission

Tasks

Test: forward turn mavic2pro X

Test: backward turn mavic2pro X

Task name Add new task

Simpleactions

1	Go forward	5	X
2	Go forward	10	X
3	Turn left	5	X
4	Turn right	5	X

Search for simpleaction▼

- You can change the order of when the robot will go forward by drag and dropping the simpleaction up and down
- After you happy with your changes, you can try to run the test mission again, and see that the robot will go forward
- You can add more simpleactions and play around with the test mission, if you want and try to see how they work

Creating a new mission

- You can now try to create an own mission.
 - Start by typing in the name of that mission in the box at the top that say 'mission name' and press 'add new mission'.
 - You will now have a mission with no tasks or simpleactions. To add a new task, do the same thing as with mission, but under 'task name' and 'add new task'. You will now have a task with no simpleactions, and you can add simpleactions in the same way as described above.

- You can create the mission by adding multiple tasks and then add simpleactions (either drag and drop or search), and then press the task allocation button to get a proposed best robot distribution for your mission, which you can manually change if you would like to.
- You can try to run your mission and change it as much as you'd like
- Finally, you can try to create a mission where two robots communicate. The implemented simpleactions for the system only allow for sending the current location of the 'mavic2pro'-drone to the 'moose'-rover.
 - To do this, add a new mission, and add two new tasks (names could be 'send location' and 'receive and go to location')
 - In the task 'send location' you first need to add a simpleaction that *set the message target* (and the target is 'moose'). Then you need to add a simpleaction that *sends the location*
 - In the task 'receive and go to location' you need to add a simple action that *receives locations from robot*. Then you need a simpleaction that specifies that the robot need to *go to location*, which in this case needs to contain two empty brackets "[]", because we get the location from the drone.
 - After you are done, the mission should look something like this:

Missions

Quarantine delivery Scout location and deliver item Scout area and deliver item Testmission **New mission**

Mission name Add new mission

Tasks

Send location	mavic2pro	X
Receive and go to location	moose	X

Task name Add new task

Simpleactions

1	Set message target	moose	X
2*	Send location		X

Search for simpleaction ▼

Show mission timeline

Send mission Automatic task allocation

- If everything was done correctly, you can open the mission timeline and see the connection between the 'send location' and 'receive location from robot' simple actions in the mission timeline. If you run the mission, then the moose should try to get to the location of the drone

- The moose will only drive around the box, because the location of the drone is right next to it, and the moose is programmed to go a few meters south of the drone, when given a location. You can try to move the drone first by adding other movement simpleactions, and then make the rover move to it.

After you are done, you can answer the questionnaire that is given. Thank you for participating!