User Manual

Remotely Operated Underwater Vehicle

Version 0.4

Author: Marcus Homelius
Date: December 13, 2017

Status

Reviewed	Amanda Andersson	2017-12-13
Approved		

Course name:	Automatic Control Project Course	E-mail:	tsrt10rov2017@gmail.com
Project group:	ROV2017	Document responsible:	Marcus Homelius
Course code:	TSRT10	Author's E-mail:	marho949@student.liu.se
Project:	Remotely Operated Underwater Vehicle	Document name:	UserManual.pdf

Project Identity

Group E-mail:	tsrt10rov2017@gmail.com
Homepage:	http://www.isy.liu.se/edu/projekt/tsrt10/2017/rov/
Orderer:	Jonas Linder, Linköping University **Phone:** +46 13 28 28 04 **E-mail:** jonas.linder@liu.se
Customer:	Rikard Hagman, Combine Control Systems AB **Phone:** +46 72 964 70 59 **E-mail:** rikard.hagman@combine.se
Course Responsible:	Daniel Axehill, Linköping University **Phone:** +46 13 28 40 42 **E-mail:** daniel@isy.liu.se
Project Manager:	Amanda Andersson **Phone:** +46 76 843 40 79 **E-mail:** amaan181@student.liu.se
Advisors:	Kristoffer Bergman, Linköping University **Phone:** +46 73 847 31 51 **E-mail:** kristoffer.bergman@liu.se

Group Members

Name	Responsibility	Phone	E-mail (@student.liu.se)
Amanda Andersson	Project manager	+46 76 843 40 79	amaan181
Marcus Homelius	Documentation	+46 70 245 90 96	marho949
George Jajji	Design	+46 70 790 16 17	geoja551
Martin Johannesson	Hardware	+46 76 949 79 69	marjo790
Mattias Mucherie	Information	+46 76 238 53 06	matmu715
Fredrik Nilsson	Software	+46 73 575 49 11	freni169
Anton Nordlöf	Modelling & Simulation	+46 76 145 04 90	antno848
Alaa Saeed	Tests	+46 76 207 57 86	alasa433

Document History

Version	Date	Changes made	Sign	Reviewer(s)
0.1	2017-11-23	First draft.	AA, MJ, MM, MH	AA, FN
0.2	2017-12-04	Second draft.	AA, MJ, MM	AA
0.3	2017-12-06	Third draft.	AA	AA
0.4	2017-12-13	Fourth draft.	AA, AS	AA

Contents

Notations

GUI Graphical User Interface

ROV Remotely Operated Underwater Vehicle

ESC Electronic Speed Controller

1 Software Requirements

Below follows a list of software and packages that are needed to start and operate the ROV.

- Ubuntu 14.04
- ROS Indigo Igloo
- rosserial (Indigo)
- rosserial_arduino (Indigo)
- image_common (Indigo)
- image_transport (Indigo)
- joy (Indigo)

2 Batteries and Battery Maintenance

In this section it is described how the batteries shall be maintained and how to connect and disconnect the battery. It will also be described how to charge the batteries correctly.

The ROV use the battery model Zippy compact 5000mAh 4S 25C Lipo Pack. This is a four cell battery with a discharge capacity of 25 Ampere. With this battery it is possible to run all thrusters without any disruptive voltage drop.

The battery power connectors are modified to HXT-4 mm connectors. The balance plug is of type JST-XH.

- The lower limit of each of the battery cells is 3.7 V. If the battery is used below this limit the battery can be harmed. The battery voltage can be checked by connecting the usb adapter to the battery balance plug as shown in Figure 4. Notice that the black cable shall be connected to the minus pin, marked with a "-". The voltage of each battery cell is displayed on the USB adapter display, see Figure 1.
- To charge the battery, connect the balance plug and the power connector to the charger. The charger has multiple type of balance plug inputs, but the one that shall be used is marked with "four cells". Then connect the black cables with each other and the read cables with each other. Figure 2 shows how the battery shall be connected to the charger. The battery shall always be supervised during charging, due to risk of fire. When the battery is fully charged, the charger will beep and all four lights will turn green.

Figure 1: The USB adapter reads the voltage for each of the four cells continuously. The voltage shall never be lower than 3.7 V for any cell.

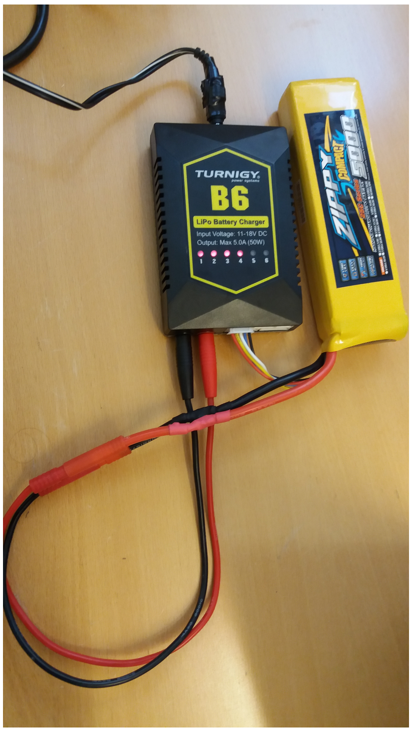

Figure 2: The connection for charging the battery.

3 Starting Up the ROV

This section describes how to start and operate the ROV.

3.1 Hardware Setup

1. Mount the battery in the bottom front of the ROV electronic cradle with velcro tape as shown in Figure 3.

2. Connect the USB adapter to the battery balance plug. Make sure that it is connected correctly, see Figure 4 for mounting description.

3. Connect the cable between the Raspberry Pi and the USB adapter as shown in Figure 5. This cable distribute power to the Raspberry Pi and the HKPilot.

4. Connect the battery power cable to the ESC:s as shown in Figure 6. If it is connected correctly, "heartbeats" will sound. Heartbeats indicates that the ROV has connection with all thrusters when no code is running on the Raspberry Pi.

5. Remove the valve on the ROV tube cap to facilitate an easy mounting of the tube cap as shown in Figure 7. If needed, add some silicon grease to the sealing on the tube cap.

6. Slide the electronic cradle into the ROV tube as shown in Figure 8. Make sure that the USB adapter display can be viewed from the outside and that no cables are exposed to stress or are clamped. Slide the cap into the tube and then insert the valve on the cap again.

7. In order to get the center of gravity at the same point each time it is important that the ROV's tube is in the correct position in relation to the ROV. How it shall be positioned is shown in Figure 9. Fixate the ROV tube to the ROV with velcro tape as shown in Figure 10.

8. Finally, connect the ROV's ethernet cable to the workstation.

Figure 3: Battery mounted in the bottom front of the electronic cradle.

Figure 4: Connection of the battery balance plug to the USB adapter. The black cable shall be connected to the minus pin, marked with a "-".

Figure 5: Connecting the cable between the Raspberry Pi and the USB adapter.

Figure 6: Connecting the battery power cable to the ESC.

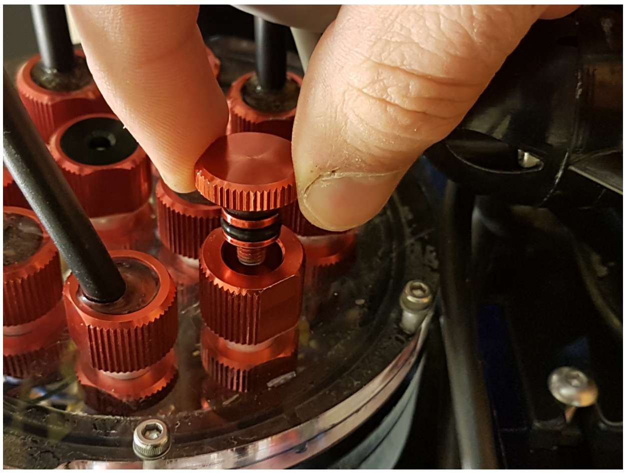

Figure 7: Removing the valve on the ROV's tube cap.

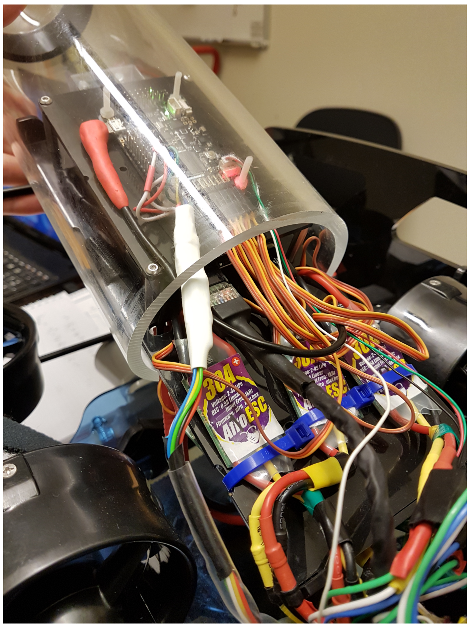

Figure 8: Sliding the electronic cradle into the ROV's tube.

(a) Overwiev of the ROV's tube position.

(b) Detailed view of the ROV's position. Notice how the tube cap is aligned with the thruster, see the yellow lines.

Figure 9: Positioning of the ROV's tube in relation to the ROV.

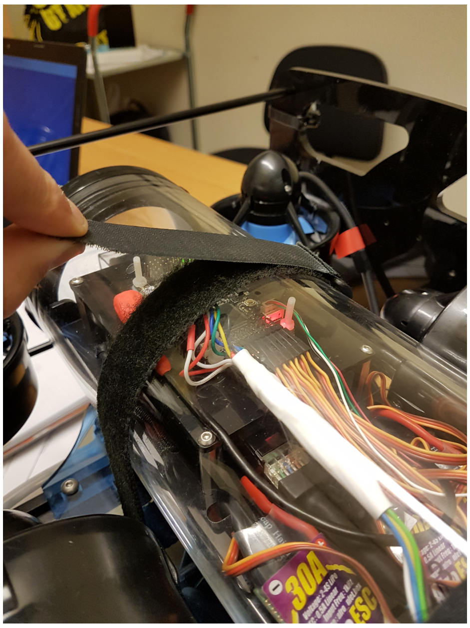

Figure 10: Attaching the ROV's tube with velcro tape.

3.2 Software Setup

When opening a new terminal window, navigate to the *catkin_ws* folder and run the command

```
source devel/setup.bash
```

This has to be done in every new window that will be opened.

To start the ROV, run the script

```
./startrov.sh
```

in a terminal on the workstation. If the connection is found, the "heartbeats" will then stop. Make sure the Xbox-controller is connected with the workstation before continuing. Open a new terminal, navigate to *catkin_ws*, and then run the script

```
./startworkstation.sh
```

This will open the GUI in a new window on the workstation.

3.2.1 Calibration

Calibration of the different sensors is made individually. To do this, open a new terminal window and navigate to *catkin_ws*.

To calibrate the gyro, place the gyro still on the ground and run the script

```
./calibrate_gyro.sh
```

and hold still for 5 seconds.

To calibrate the accelerometer, run the script

```
./calibrate_accelerometer.sh
```

Instruction will be shown in the first terminal. Turn the ROV according to the instruction and hold still. A new instruction will be shown after 7.5 seconds. When all sides have been hold up, the calibration is done. The calibration of the accelerometer can with advantage be done in the water.

To calibrate the magnetometer, run the script

```
./calibrate_magnetometer.sh
```

and then move the ROV in circles while rotating it for 30 seconds. The calibration of the magnetometer can with advantage be done in water.

To calibrate the sonars,put the ROV in the pool and run the following scripts

```
./sonar_deactivate.sh
./sonar_activate.sh
```

This will calibrate the sensors for measurements in water.

To initialize the magnetometer values used in the sensor fusion module and initialize the bias of the gyro, run the following script when the ROV stands still

```
./calibratesensors.sh
```

The ROV should be aligned with the PCS coordinate system when running the script.

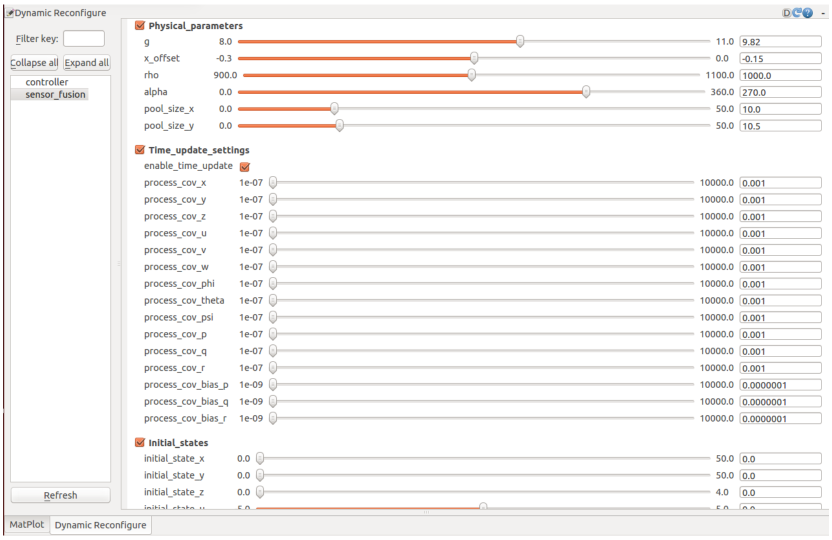

Figure 11: Example of how to set the pool size and initial condition for the ROV.

4 Shutting Down the ROV

To shutdown the ROV, use **Ctrl+C** in all open terminals to kill all active nodes. To ensure the SD-card in the Raspberry Pi are not damaged, run the script

```
./shutdownrasp.sh
```

to shut it down correctly.

5 Operating the ROV

There are currently three different modes that can be used to operate the ROV in. The first mode is a manual control of the ROV with a Xbox-controller. The second mode is a Xbox mode where velocity references is sent from the Xbox-controller. The third mode is a reference mode, where the reference can be set on the GUI map for the pathfinder.

Before operating the ROV, the pool size and the start position for the ROV has to be set, to ensure that the sensor fusion is able to work correctly. In the GUI, go to Dynamic Reconfiguration and select sensor_fusion in the left column. Scroll down to Physical_parameters (see Figure 11). Set the parameters pool_size_x and pool_size_y to the dimension of the pool. Then go to initial_states and set initial_state_x, initial_state_y and initial_state_z to the ROV's start position.

When the pool size and the initial condition is set, open a terminal window, navigate to *catkin_ws* and then run the following script to restart the filter:

```
./restartFilter.sh
```

5.1 Manual Mode

In the manual mode, the ROV is controlled manually with the Xbox-controller. A description of how the different buttons and triggers can be used to control the ROV in Xbox mode is shown in Table 1.

Button/Stick	Description
A-button	Enable the thrusters.
B-button	Disable the thrusters.
Left stick	Velocity references in the ROV's x- and y-direction.
Right stick	Angular velocity references in pitch and yaw.
Left trigger	Increase depth velocity reference.
Right trigger	Decrease depth velocity reference.
Left bumper	Positive roll rate reference.
Right bumper	Negative roll rate reference.

Table 1: The different buttons and triggers on the Xbox-controller and a description of their functionality.

To activate the manual mode, go to Dynamic Reconfiguration in the GUI and select controller in the left column. In the controller drop-down menu, enable the option ManualControl and in the xbox drop-down menu, enable the option Yes, see Figure 12.

5.2 Xbox Mode

The Xbox mode resembles the manual mode, but the velocity references from the Xbox-controller is sent to, and controlled by, the velocity controller.

To activate the Xbox mode, go to Dynamic Reconfiguration in the GUI and select controller in the left column. In the controller drop-down menu, enable the option LinVelAngVel and in the xbox drop-down menu, enable the option Yes. The velocity controller can be tuned by adjusting the trimming parameters between 10^{-6} and 10^{6}, see Figure 13.

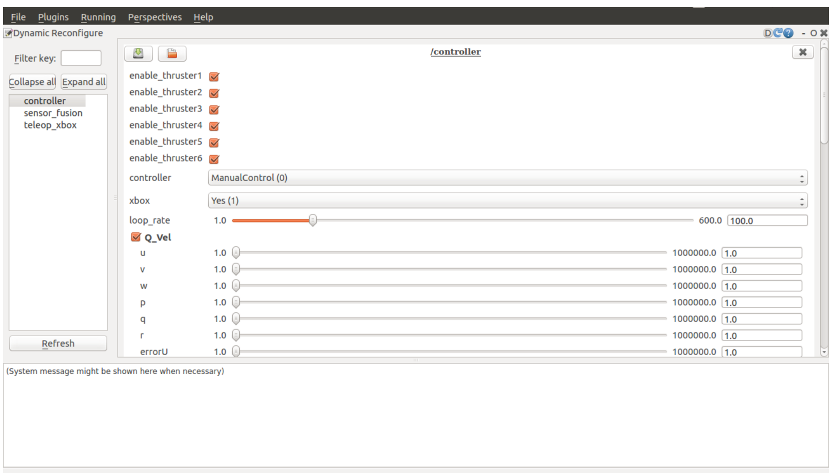

Figure 12: Example of how to set to manual mode in the GUI, in the Dynamic Reconfiguration.

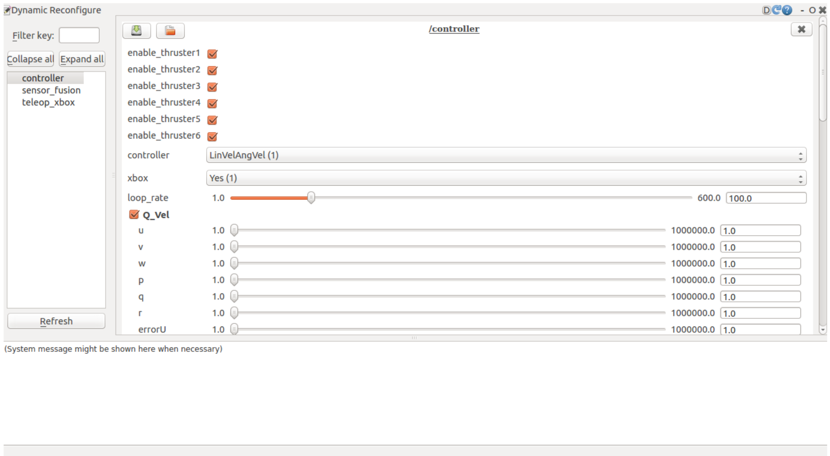

Figure 13: Example of how to set to Xbox mode in the GUI, in the Dynamic Reconfiguration.

5.3 Reference Mode

In reference mode, a desired reference point is set from the GUI. A path is then set that the ROV will follow to reach the desired point.

To activate the reference mode, go to Dynamic Reconfiguration in the GUI and select controller in the left column. In the controller drop-down menu, enable the option PosAtt and in the Xbox drop-down menu enable the option No. The position and attitude controller can be tuned by adjusting the trimming parameters between 10^{-6} and 10^{6}, see Figure 14.

Also set the dimensions of the pool for the visualization. To change pool dimensions, go to Dynamic Reconfiguration in the GUI and select grid_map_rov_map in the left column. Adjust the pool dimensions and also the resolution of the map by changing the cell width, see Figure 15. Note that setting a low cell width can make the visualization run slowly depending on your system. The default and also recommended cell width is 0.1 m. When changing the cell width, the visualization part of the GUI (RViz) needs to be updated as well. In the Display panel, under "PointCloud2", update the parameter "Size (m)" to fit the set cell width. The default value is here again set to 0.1.

In the map on the GUI, the start position can be set by clicking on "2D Pose Estimate" in the header menu and then click on the map to set the start point. See Figure 16. Preferably, set the start point to the ROV's real position. If the start point is not the ROV's real position, the ROV will navigate to the starting position without taking any obstacles or walls into account. The end position is set by choosing "Publish Point" and then click on the desired end position on the map. The pathfinder will then calculate the path between start and end position and visualize it on the map. The path is only sent to the controller module when the ending point has been updated. Updating the starting point will not send the path but it will update the map visually. The position of the ROV can be visualized on the GUI map to ensure that it follows the desired path.

5.4 Logging Data

To record data from a run, begin with opening a new terminal window and navigate to *catkin_ws*. To start the recording, run the script

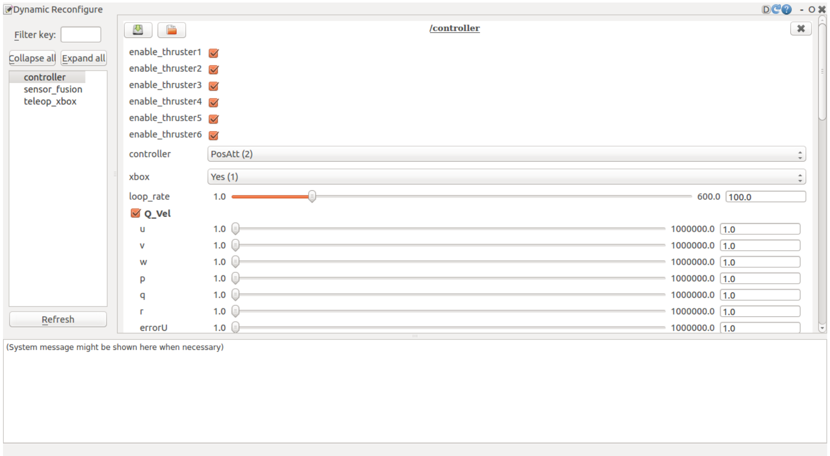

Figure 14: Example of how to set to reference mode in the GUI, in the Dynamic Reconfiguration.

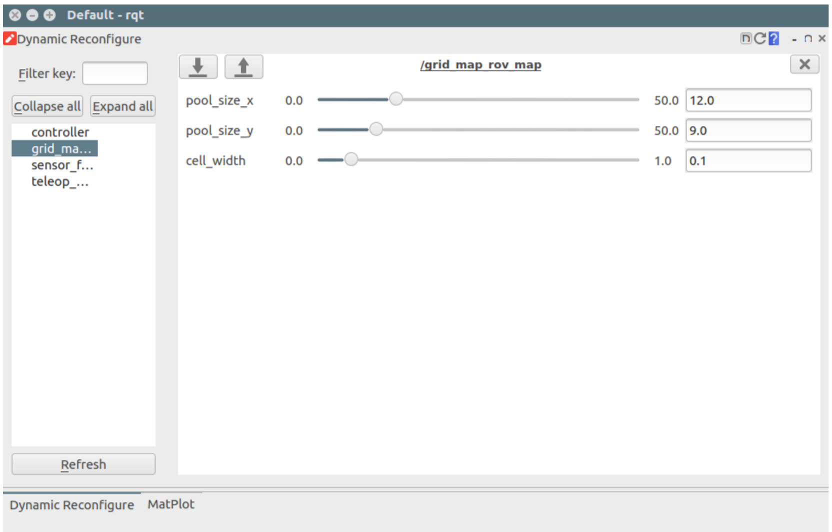

Figure 15: Example of how to set to set pool dimensions in the GUI, in the Dynamic Reconfiguration.

```
./log.sh
```

This will enable logging of the topics rovio/thrusters, sensor_fusion/states, reference, rovio/enable_thrusters rovio/water_pressure/data, rovio/magnetometer/data rovio/imu/data, rovio/sonar1/data, rovio/sonar2/data and rovio/sonar3/data. Before the recording start, it is possible to name the log file. To end the data recording, press **Ctrl+C** twice. The logged data is saved as a bag file in the map ../testdata/bag.

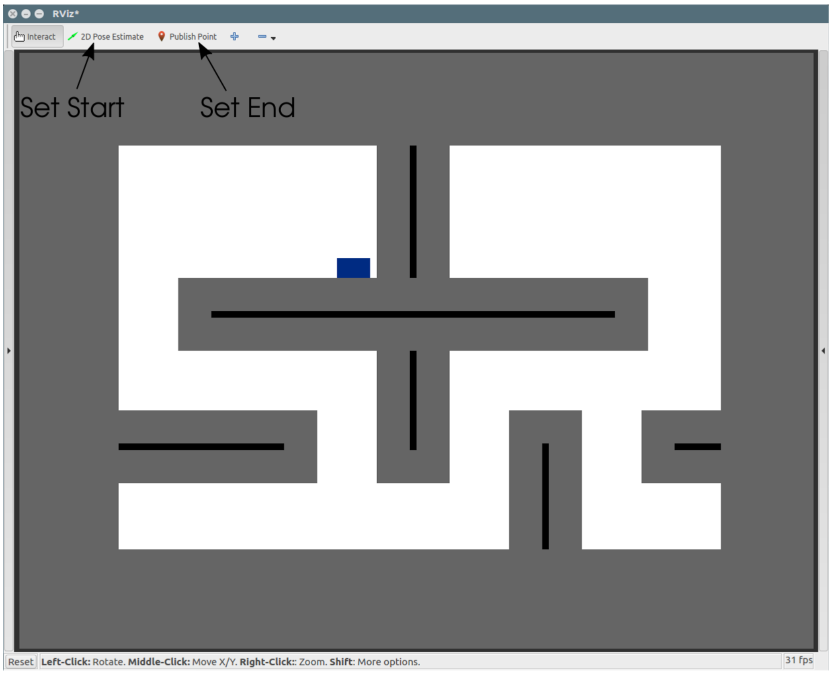

Figure 16: Example of how the visualization part of the GUI looks like and the button needed to use it.

5.5 Displaying the Continuous Plots

To plot data that publishes on different topics, the MatPlot on the GUI can be used. To display the states, write /sensor_fusion/states[0] and to display the reference, write /reference/data[0]. The index number in the hard brackets denote the different data that will be shown, see Table 2 and 3.

6 Further Development of the ROV

To build newly added or changed code, run the command:

```
cd rov/catkin_ws && catkin_make
```

The ROS package that runs the arduino is the *bluerov_arduino_firmware_rovio*. To compile and flash this onto the arduino run

```
cd rov/catkin_ws && catkin_make bluerov_arduino_firmware_rovio &&
↪  catkin_make bluerov_arduino_firmware_rovio-upload
```

Index	Description
0	x-position in PCS.
1	y-position in PCS.
2	z-position in PCS.
3	Roll angle of LCS in PCS.
4	Pitch angle of LCS in PCS.
5	Yaw angle of LCS in PCS.
6	Linear velocity in LCS x direction.
7	Linear velocity in LCS y direction.
8	Linear velocity in LCS z direction.
9	Angular velocity around LCS x-axis.
10	Angular velocity around LCS y-axis.
11	Angular velocity around LCS z-axis.
12	Depth velocity

Table 2: Description of the content of the messages' different index number sent on /reference/data.

Index	Description
0	x-position in PCS.
1	y-position in PCS.
2	z-position in PCS.
3	Linear velocity in LCS x direction, u.
4	Linear velocity in LCS y direction, v.
5	Linear velocity in LCS z direction, w.
6	Roll angle of LCS in PCS, ϕ.
7	Pitch angle of LCS in PCS, θ.
8	Yaw angle of LCS in PCS, ψ.
9	Angular velocity around LCS x-axis, p.
10	Angular velocity around LCS y-axis, q.
11	Angular velocity around LCS z-axis, r.
12	Bias state for p.
13	Bias state for q.
14	Bias state for r.

Table 3: Description of the content of the messages' different index number sent on /sensor_fusion/states.