POLITECNICO
MILANO 1863

School of Industrial and Information Engineering

MSc in Mechanical Engineering

Autonomous Vehicles

2022-2023 Academic Year

Assignment 1

Isabel Pollini

Introduction

The aim of this assignment is performing an analysis of the turtlebot behaviour recorded in the ‘ex_A1.bag’ bag file by using Matlab&Simulink. The environment in which the bot was moving presented obstacles. The requests for this assignment are:

- Evaluation of the minimum distance with respect to obstacles at each time.
- Estimation of the /cmd_vel (command velocity) sequence provided to the robot.
- Writing a publisher with the estimated sequence and save a bag containing the published commands.

Results

This assignment could be performed either on Simulink or Matlab: the proposed solution is done only in Matlab. The code used to perform the analysis is here reported.

```
% Read messages from bag
selBag  = rosbag('ex_A1.bag');
scanBag = select(selBag,'Topic','/scan');
msgScan = readMessages(scanBag,'DataFormat','struct');
odomBag = select(selBag,'Topic','/odom');
msgOdom = readMessages(odomBag,'DataFormat','struct');

% Defining vector
scanMins    = zeros(length(msgScan),1);
velLinX     = zeros(length(msgOdom),1);
velAng      = zeros(length(msgOdom),1);

% Time vector definition
odomTime    = odomBag.MessageList.Time;
scanTime    = scanBag.MessageList.Time;

% Getting the required information
for ii = 1:length(msgScan)
    scan_it         = msgScan{ii,1};
    scan_xy         = rosReadCartesian(scan_it);
    x               = scan_xy(:,1);
    y               = scan_xy(:,2);
    scanMins(ii,1)  = min(sqrt(x.^2+y.^2));
end

for ii = 1:length(msgOdom)
    velLinX(ii,1)   = msgOdom{ii,1}.Twist.Twist.Linear.X;
    velAng(ii,1)    = msgOdom{ii,1}.Twist.Twist.Angular.Z;
end

% Plots
figure
plot(odomTime,velLinX)
```

```
hold on
grid on
title('Linear Velocity')
xlabel('Time [s]')
ylabel('Velocity [m/s]')

figure
plot(odomTime,velAng)
hold on
grid on
title('Angular Velocity')
xlabel('Time [s]')
ylabel('Velocity [rad/s]')

figure
plot(scanTime,scanMins)
hold on
grid on
title('Minimum distance from obstacles')
xlabel('Time [s]')
ylabel('Distance [m]')
```

To obtain the requested quantities, the scan and odom topics have to be extracted from the bag, to then read and analyse their messages.

To calculate the minimum distance to the obstacles, the rosReadCartesian function can be used, in order to convert the scanned position vectors into cartesian points. The distance to each of these points and the robot is evaluated and the minimum is found for every time step. The results are shown in Figure 1.

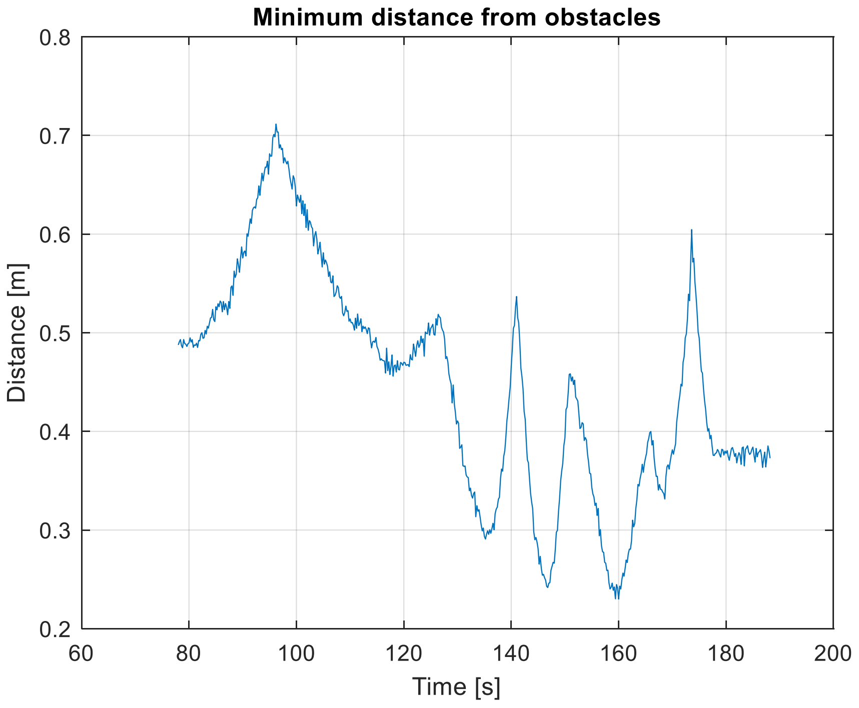

Figure 1. Minimum distance of turtlebot from the obstacles evaluated at any time instant.

The second request consists in estimating the original command velocities, imposed in the cmd_vel topic. The input is divided in the two messages: one indicating the linear velocity in X direction in the bot's reference system and the angular velocity around the Z axis. Figure 2 shows the linear velocity, with Figure 3 reports the angular velocity, both plotted versus time.

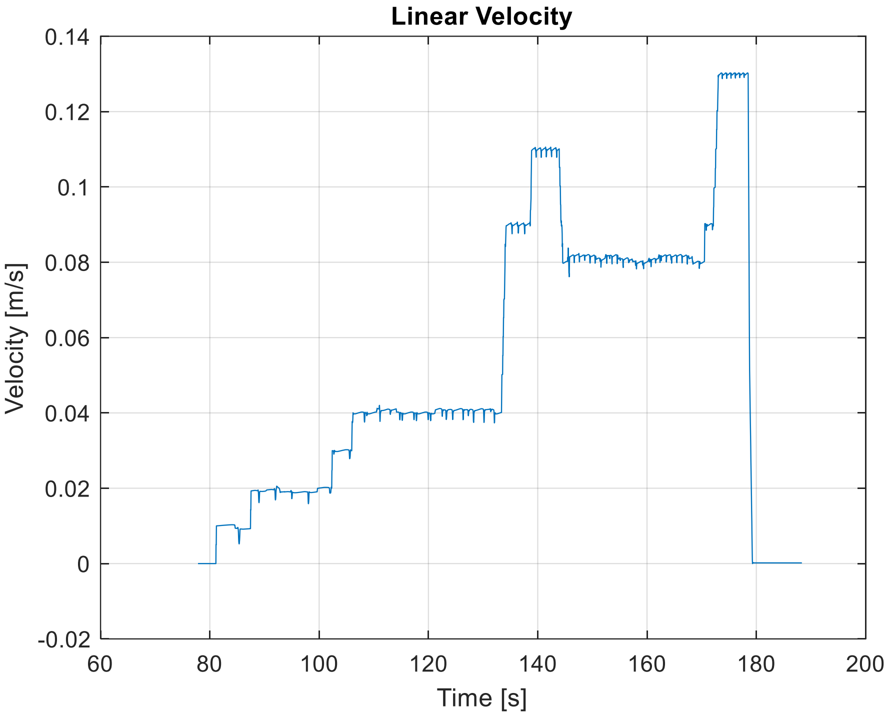

Figure 2. Linear velocity imposed to the turtlebot evaluated at any time instant.

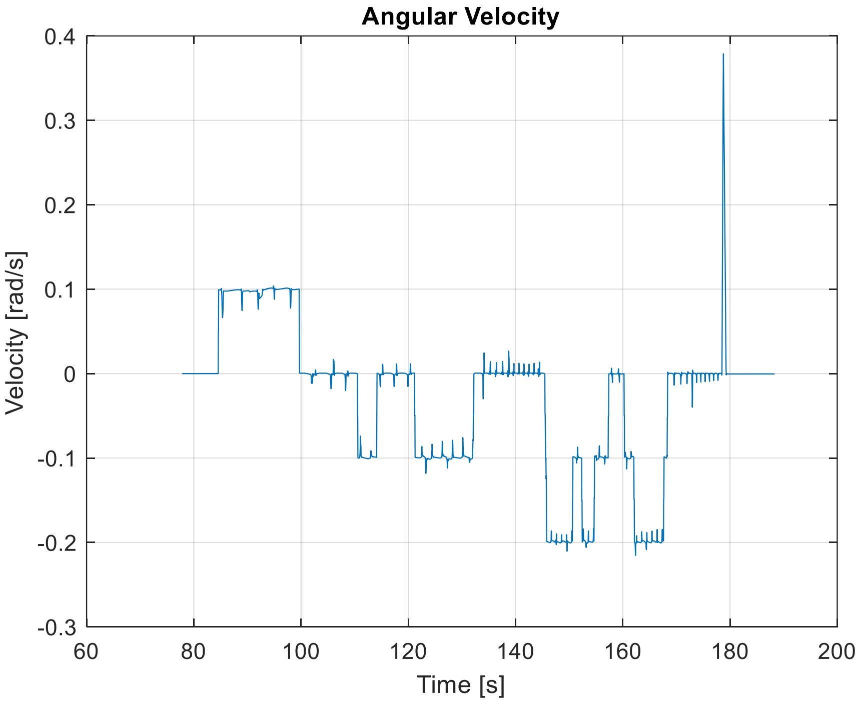

Figure 3. Angular velocity imposed to the turtlebot evaluated at any time instant.

The found velocity vectors are saved into a matrix called ‘speedMatrix’, that is used in the publisher to recreate the original commands. The code of the publisher is here reported.

```
load('speedMatrix.mat')
tbot                        = turtlebot('localhost');
n                           = length(speedMatrix);

tbot.Velocity.TopicName = '/cmd_vel';
desiredRate                 = 30;
rate                        = rateControl(desiredRate);

for ii = 1:n
    setVelocity(tbot,speedMatrix(ii,1),speedMatrix(ii,2))
    waitfor(rate);
end
```

Bonus Request Results

This assignment also presented a bonus request, which consisted in evaluating the accuracy of the estimated velocities.

This was obtained by running the bot with the Matlab publisher written at the end of the first request and saving a bag containing the data.

The following plots show the comparison between the original and the estimated signal for both linear and angular velocities, and the error, calculated as the difference between the original and the estimated signal. The initial time of the signal has been superimposed to evaluate eventual delays.

Its can be seen that, although the shape is very similar, the estimated signal anticipates the original one, with a delays that grows with time. This is probably due to the approximation of the rate in the publisher.

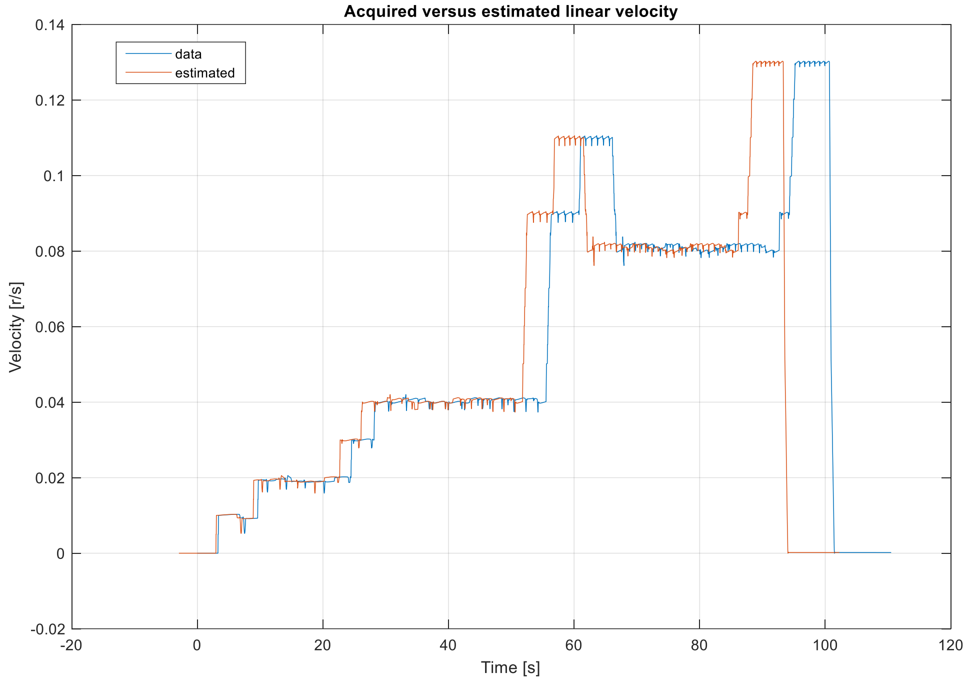

Figure 4

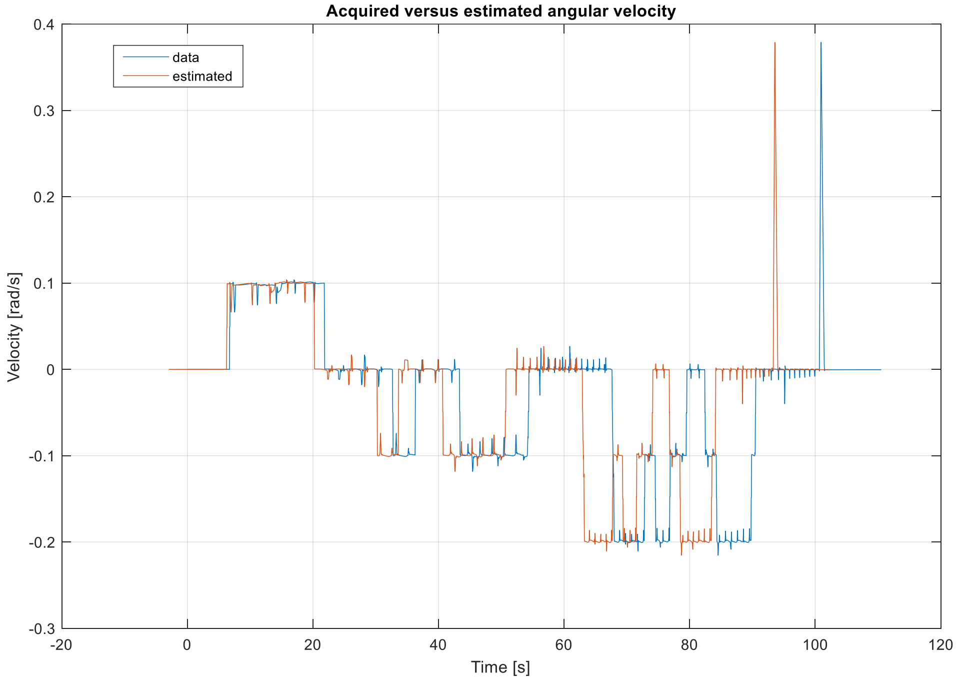

Figure 5

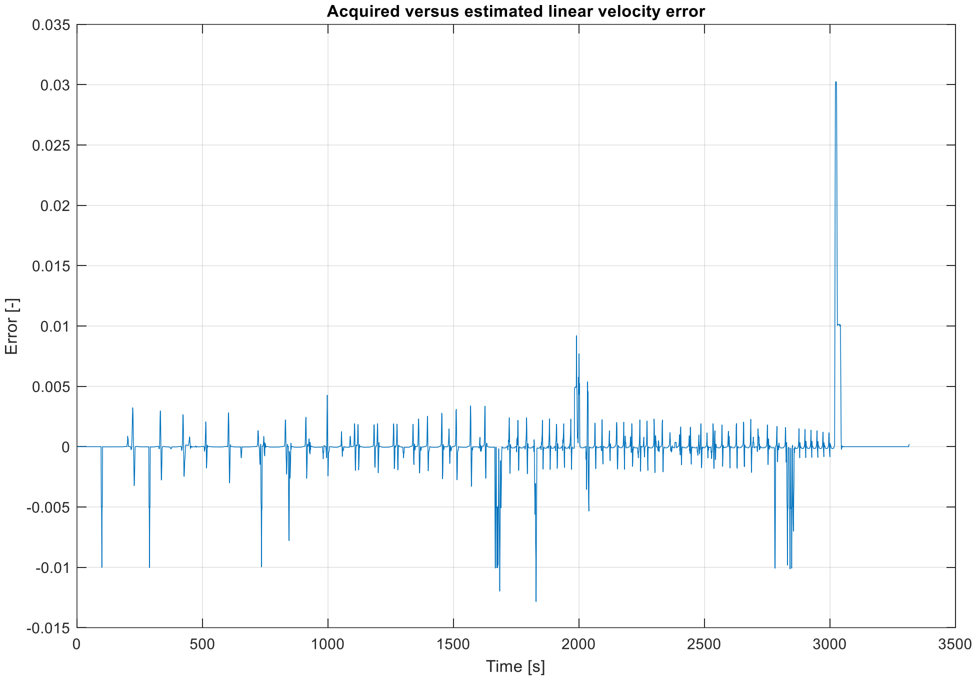

Figure 6

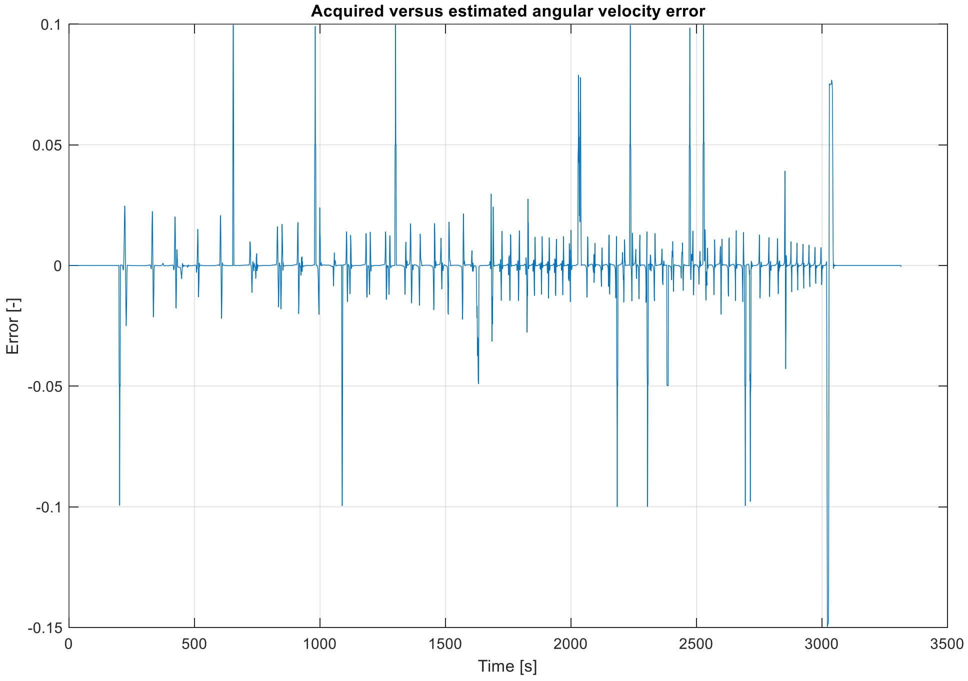

Figure 7

Conclusion

The analysis of the given bag was performed and the requests of the assignments have been satisfied.