1 Introduction

This laboration is about localisation using a network of sensors, a problem that is common in the sensor fusion world. A real world example of audio based localisation is shooter localisation. The sound of a gun being fired is quite distinctive and can be correlated with the recordings. Time of arrival at different microphones can be found, and the shooter can hence be located. Another example is localisation of airplanes using radars.

In this laboration you will use a network of microphones to localise a target that is emitting short sound pulses. To simplify things, the profile of the pulse that is being emitted is known. Hence, correlation can be performed with the recordings to find the time of arrival (TOA) of each pulse at each sensor. If the time of pulse emission is known, one can easily find the distance to the target using the (also known) speed of sound. This gives a radial distance to the target, but not a bearing, and thus the target location could be anywhere on a circle (assuming a 2D world). Combining such circles for all the sensors, the intersection, which is where the target is located, can be found.

In this laboration, as in most real world cases, the time of emission is not known. When all sensors are synchronised, one method of performing localisation when time of emission is unknown, is to use time difference of arrival (TDOA). Knowing the TDOA, one can use the speed of sound to find the relative difference in distance between sensors and the target. Given all relative distances in the network, the location of the sound source can be found. For further information on TOA and TDOA, see Chapter 4 in [1].

2 Purpose of laboration

During the laboration a LEGO® MINDSTORMS robot will autonomously follow a planar, closed loop track, while simultaneously repeatedly playing a sound pulse, see front cover. The purpose of the laboration is to track a sound-emitting robot by recording audio with seven microphones providing TDOA measurements.

3 Equipment

The equipment used is briefly explained below.
Sound Card

The sound card is a *Saffire PRO 10 I/O*, see Figure 1, which allows up to 10 channel recording and 10 channel playback simultaneously. Please refer to the *User Guide* available at the laboration site for more information about the hardware.

The sound card has a software interface called *SaffireControl PRO* (icon located on the desktop) which basically has the same functionality as the sound card itself, but also enables more settings. *SaffireControl PRO* shows if the sound card is connected correctly. You may have to restart the *Saffire-Control PRO* if the sound card cannot be found.

![Saffire PRO 10 I/O sound card](image)

(a) Front  (b) Rear

**Figure 1:** *Saffire PRO 10 I/O* sound card.

Additional audio software

The software *Playrec* from [http://playrec.co.uk](http://playrec.co.uk) is designed for simultaneous multi-channel recording and playback. It has been installed on the computers and makes it possible to use the *Saffire PRO 10 I/O* within MATLAB, see Section 5. There is no need to use *Playrec* directly at the laboration.

Microphones

There are seven microphones used for recording. Make sure that the microphones are switched on when recording!
Target

The target in this laboration is a LEGO® MINDSTORMS robot, see Figure 2, which uses a light sensor to detect the different colors of the tapes that constitute the trajectory. The robot comes with pre-installed programs, see Section 6. For example, SFdrive makes the robot follow the taped lines while playing the sound signal described in Section 4.

![Figure 2: LEGO® MINDSTORMS robot.](image)

Environment

The laboration environment, see Figure 3, is a 0.991 × 1.222 m² wooden board with three colors of tape in a loop. The loop shape makes it possible to run the trajectory several times.

![Figure 3: Robot environment.](image)
TDOA Quality Estimation

On the other side of the wooden board, a circle arc with radius 0.7 m is indicated. Place the microphones and the robot similar to Figure 4 when collecting data for TDOA variance and bias estimates. This orientation of the robot ensures that its speaker is facing the microphones.

Figure 4: Suggested setup when estimating the quality of the TDOA measurements.

4 Signal Processing

The robot emits a pulse train, where each pulse is an orthogonal frequency-division multiplexing (OFDM) signal generated to be insensitive to noise disturbances. The pulse $p(t)$ is generated from a finite number of discrete frequencies $P(f)$. An example of such a signal is shown in Figure 5. Since the emitted signal is known, it is possible to correlate the measured signals of the microphones with a copy of the pulse. This is the autocorrelation and it will have quite distinct maxima. The maxima which will be the TOA on every channel. There will also be other maxima due to noise, echos etc. but these are hopefully less distinctive than the interesting peak. Since the robot is emitting a sequence of pulses, several maxima will be found. Even in situations with a low Signal to Noise Ratio (SNR), this method performs quite well.

In the laboration the robot will play a signal with pulse width $t = 0.1$ s made of frequencies between $f = 800 - 1200$ Hz. The time between pulses is
approximately 0.5 seconds.

5 Provided code

The following MATLAB functions are provided:

- **SFlabRun** — the main script for recording and preprocessing the data.
- **SFlabPlayAndRecord** — simultaneously plays sound on up to eight output channels and records from up to eight input channels.
- **SFlabPlotRecData** — plots the results from a recording session.
- **SFlabPlayRecData** — plays the recorded data from either all input channels, or from a user specified subset of the input channels.
• SFlabFindPulseTimes — finds the times at which pulses were detected in each input channel. These times can be used for TDOA localisation.

• SFlabCompEstimGroundTruth — plots the microphone positions and the trajectory estimates overlaid on the robot environment.

Type `help` followed by the function name in the MATLAB command prompt to see how to use the functions. A number of auxiliary functions are also provided, however, they are only used by the functions above and need not be called by the user. The MATLAB functions can be downloaded from the course homepage.

6 Robot functions

The following Not Exactly C (nxc) functions are provided and stored in the LEGO® MINDSTORMS robot:

• SFdrive — makes the robot follow the trajectory while playing the sound.

• SFpulse — plays the sound described in Section 4 while the robot stands still.

• SFdriveSilent — makes the robot follow the trajectory without playing the sound.

The functions are started like this:

1. Press the orange button to start the robot’s computer.

2. Select My Files and press the orange button.

3. Select Software Files and press the orange button.

4. Select the function of your choice and press the orange button.

To interrupt a function, press the gray rectangular button. Place the robot centered above the tape trajectory, in the clockwise direction, before the line following programs are started.

The robot runs on a battery. It is a good idea to charge the battery in-between experiments, since the line following capability degrades when the battery runs low.
7 Tasks

The laboration is performed in groups of two students. Data acquisition is done in the lab, where all the necessary equipment is located. One 30 min time slot is provided for each group, which can be booked via the lab registration system. The remainder of the laboration is performed elsewhere.

7.1 Data collection

The first part of the lab is data collection. Perform the following tasks:

1. Use the suggested setup in Section 3 to obtain data to estimate the measurement precision.

2. Let the robot follow the trajectory, while playing the sound. Record data with at least two different microphone configurations. Do not forget to note the microphone positions. The two different configurations should correspond to fundamentally different situations. You should plan your configurations before the lab session. Record 45 s of data for each configuration (this corresponds to the robot driving the track approximately twice). Note the starting point of the robot, it can be used to initialise the localisation.

3. Save all your recorded data using the save command in MATLAB.

After a data recording it is important to verify that the recording was successful, here the commands SFlabPlayRecData and SFlabPlotRecData are useful. For each data set the command SFlabFindPulseTimes should be used to estimate the time of arrival for each pulse and each sensor. It is a good idea to run this command during the laboration to verify that all pulses are detected successfully.

7.2 Localisation

When data has been recorded, the remainder of the lab should be performed elsewhere. The following tasks should be solved:

1. **Sensor calibration:** Describe the accuracy of the measurement (the estimated time of arrival) for each sensor in terms of the bias and standard deviation of its measurement noise. Use the matrix $\text{tphat}$ from the calibration data to compute a vector of measurements errors $\mathbf{e}$ for each sensor (notice that this can be accomplished in different ways
since the true time of arrival is unknown). These measurement errors
can be visualized in a normalized histogram

\[ [N, l] = \text{hist}(e, 20); \]
\[ Wb = l(2) - l(1); \% \text{Bin width} \]
\[ Ny = \text{length}(e); \% \text{Nr of samples} \]
\[ \text{bar}(l, N/(Ny*Wb)); \]

which can be seen as an empirical probability density function of the
measurement noise. Compare the histogram with a pdfclass distribu-
tion, for instance a normal distribution \( pe = \text{ndist}(\text{mean}(e), \text{var}(e)) \),
where \( \text{mean}(e) \) is the bias and \( \text{sqrt}(\text{var}(e)) \) is the standard deviation
of the measurement noise. This distribution can be visualized using
\( \text{plot}(pe) \).

2. **Signal modeling**: Describe the different sensor models you will use
in your localisation algorithms. The models should be given both as
equations

\[ y = h(x) + e, \quad e \sim \mathcal{N}(0, R) \]

and as an m-file.

3. **Experiments**: Write a brief description of the microphone configu-
   rations, and the obtained data. The sensor locations and the target’s
   initial position can, together with the m-file and distribution \( pe \) above,
   be used to construct a \texttt{sensormod} object. The configurations can then
   be illustrated with \texttt{sensormod.plot}.

4. **Configuration analysis**: Compare your two configurations using at
   least one of your sensor models in at least one of the following aspects:

   (a) Compute the nonlinear least squares (NLS) loss function
   \[ V(x) = (y - h(x))^T R^{-1}(y - h(x)) \]
on a grid over the table for a certain
   measurement \( y \).

   (b) Compute the Cramér Rao Lower bound (CRLB) as a function of
   target position and represent this as a map over the table where
   each grid point gives a bound on the root mean square error
   \[ \text{RMSE} \geq \sqrt{\text{tr} \left( I^{-1}(x) \right)}. \]

   Use this map to motivate your preferred configuration.

5. **Localisation**: Estimate the position of the target at each time instant.
   Compare at least two of the following localisation algorithms:
(a) NLS using a 3D grid search over \( x = [p^T, r_0]^T \), where \( p \) corresponds to the position and \( r_0 \) to the unknown pulse emission time.

(b) NLS using a gradient or Gauss-Newton search over \( x = [p^T, r_0]^T \).

(c) Separable least squares (SLS) using a weighted least squares (WLS) estimate of \( r_0 \) and a 2D grid search over \( p \).

(d) A TDOA approach, where the pairwise differences of detection times for the \( M \) sensors are used to eliminate \( r_0 \). All \( M(M-1)/2 \) pairs can be used, or one sensor can be a reference, and the \( M-1 \) pairwise differences are used only.

6. **Tracking:** First, select two motion models. Then, compare the following nonlinear filters for both models (that is, you will get four estimated trajectories):

(a) Take the localisation estimates above as the artificial measurements \( y_k = \hat{p}_k + e_k, e_k \sim N(0, P_k) \) at time instant \( k \). Apply the extended Kalman filter (EKF) using the two different motion models in turn.

(b) Select one sensor as a reference. Make sure that this one gives good measurements. Use a TDOA measurement model with seven TOA differences. Apply the point mass filter (PMF), EKF, unscented Kalman filter (UKF) or particle filter (PF).

7. **Sensitivity analysis:** Select one of the methods above and evaluate how sensitive the result is with respect to the specified microphone locations. After all, these locations are measured by hand, and contain an uncertainty. What is the size of this uncertainty approximately, and will it affect the result?

8. **Lab Report**

The laboration is not examined at site, but through a written report. Check the course homepage to find out when the report is due. The report must include the following:

1. A description of how the data was gathered. This includes the microphone positions, and why the particular positions were chosen.

2. Descriptions of how you solved the tasks in Section 7.2. Include plots that illustrate your results.
3. Well supported conclusions.

Each group is given an ID by the lab assistant when collecting the measurements in the lab. This ID should be written on the lab report, do NOT write your names or personnummer on the lab report. The report shall be sent as a pdf file to the course assistant and Urkund (the e-mail addresses are stated on the course homepage). The title of the email should be **TSRT14 lab report**. Name your pdf-file according to your given ID. For example if you are given 54 as ID, you name the pdf-file `report54.pdf`. Your report will not be processed until this naming convention is enforced.

9 Peer Review

In peer review, your task is to provide feedback that will help other students to improve their report, and also to allow you to learn from seeing how other people write their report. Shortly after handing in your lab report you will receive another group’s lab report from the course assistant. Read this lab report and write a review report. Check the course homepage to find out when the review report is due. In the review report you should respond and discuss the following questions

1. Are the data sets presented clearly? Are the procedures to acquire data described in enough detail for the experiment to be repeated by someone else?

2. Is there a clear explanation of the solutions of the tasks in Section 7.2? Discuss each task 1–7 separately.

3. Are the conclusions well supported by the data, the experiment and the results? Do you agree with the conclusions? What would you like to add to the conclusions, based on the data and the results of the task?

4. What is a particular strength in this lab report? Discuss the content, not the format.

5. What suggestions can you make for improving the overall quality of the report? Discuss clarity, readability and technical accuracy.

The questions shall be answered in the form of a discussion, present arguments for your point of view and propose alternative methods. Some more specific tips:

- Present useful criticism and make sure your comments are constructive.
• Use a positive tone and consider how you would feel if someone sent your review to you.

• Be clear and specific about things you think could be improved.

• Point out strengths as well as weaknesses.

• Use a courteous language.

• Avoid comments that might be read as insulting or inappropriately personal.

Most people ignore feedback that they find hostile, vague, or confusing. Try to keep your comments positive and specific: this will make them much more useful to your peers.

The length of the review report shall be 1–2 pages, and you shall quote your group ID as well as the group ID of the lab report you reviewed. Do NOT write any names on the report. The report shall be send as a pdf-file to the course assistant (the email address is stated on the course homepage) and the title of the email should be TSRT14 review report. The naming convention for the review report is as follows. If ID1 is the ID for the group who’s report is being reviewed and ID2 is the ID for the group who is conducting the review, name the review report reviewID1byID2.pdf. Hence, if for example your group’s ID is ID2 = 54, and the ID of the group who’s report you have reviewed is ID1 = 65, then the review report should be named review65by54.pdf. Your report will not be processed until this naming convention is enforced.

Both the lab and the review reports will be read by the assistant and given one of the following grades: pass, clarifications needed, or fail. If clarifications are needed, a new version of the report must be handed in within one week of its return to the student. If the complemented version is not passed or the report is failed at first hand in, a new version will be read and graded in conjunction with the next course exam.

10 Preparations

• Read Chapter 4 in [1].

• Plan at least two different microphone configurations. The two different configurations should correspond to fundamentally different situations.

• Download SFlab1.zip from the course homepage to your student account and unzip the files.
References