Location Template Matching-based Study of Acoustic Emission Localization using e-Puck Robots

A Thesis Presented

by

Cagri Dikilitas

to

The Department of Mechanical Engineering

in partial fulfillment of the requirements
for the degree of

Master of Science

in

Mechanical Engineering

Northeastern University
Boston, Massachusetts

August 2015
To life.
# Contents

| List of Figures | vi |
| List of Tables | viii |
| List of Acronyms | ix |
| Acknowledgments | x |
| Abstract of the Thesis | xi |

## 1 Introduction

1.1 Motivation for Research ........................................... 1
1.2 Goal of Study .................................................. 1
1.3 Acoustic Source Localization Methods in Literature .............. 2
1.4 Assumptions and Limitations ..................................... 3
1.5 Steps of this Research ........................................... 3

## 2 Experimental Setup, Hardware and Software

2.1 Mobile Robot e-Puck .............................................. 6
2.2 Kistler Accelerometer ............................................ 8
2.3 Acoustic Emission Source ......................................... 8
2.4 Emission Propagation Medium - Acrylic Sheet .................... 12
2.5 Communication and Software ..................................... 12
2.6 Data Preparation .................................................. 14

## 3 Wave Motion in Elastic Solids

3.1 Wave Propagation .................................................. 17
3.2 Types of Wave Propagation ....................................... 18
3.3 Waves and Signals In Our Experiment ................................ 20
3.3.1 Waves and Signals with A Direct Hit On The Accelerometer .. 20
3.3.2 Waves and Signals Acquired on e-Puck and Kistler ............ 23
4 Triangulation
  4.1 Overview ................................................................. 25
  4.2 Details of Method ..................................................... 25
  4.3 Various Experimental Examples and Accuracy ..................... 28
  4.4 Comments ............................................................... 28

5 Signal Processing and Statistics ........................................ 30
  5.1 Random Processes ..................................................... 30
    5.1.1 Random Variables ............................................... 30
    5.1.2 Discrete Random Variables .................................. 31
  5.2 Stochastic Processes ................................................ 33
    5.2.1 Time-Dependent Random Variables ......................... 33
    5.2.2 Correlation Functions ....................................... 35

6 Correlation-based Location Template Matching ...................... 39
  6.1 Overview ............................................................... 39
  6.2 Details of CLTM Method ............................................. 39
  6.3 Algorithm Implementation ......................................... 42
  6.4 Signals on e-Puck and Kistler ................................... 44
  6.5 Results and Comments .............................................. 44

7 Group Delay-based Location Template Matching ..................... 47
  7.1 Overview ............................................................... 47
  7.2 Details of GLTM Method ............................................. 47
    7.2.1 Theoretical Definitions ...................................... 48
    7.2.2 GLTM Method .................................................... 52
  7.3 Algorithm Implementation ......................................... 54
  7.4 Results and Comments .............................................. 55

8 Model Based Location Template Matching ................................ 57
  8.1 Infinite Impulse Response Filter-based LTM ...................... 57
    8.1.1 Overview ......................................................... 57
    8.1.2 Details of Method ............................................. 57
    8.1.3 Algorithm Implementation .................................... 61
    8.1.4 Results and Comments ....................................... 62
  8.2 System Identification Based Location Template Matching ........ 63
    8.2.1 Overview ......................................................... 63
    8.2.2 Troubles Faced with Other Approaches when using e-Puck .... 64
    8.2.3 Our Approach to Deliver Localization using e-Puck with modified IIRF-based LTM ........................................... 64
    8.2.4 Algorithm Implementation .................................... 65
    8.2.5 Results ............................................................ 66
## List of Figures

1.1 General view of experimental setup ........................................... 4

2.1 e-Puck robot ................................................................. 6
2.2 Adafruit's NEMA-17 size stepper motor ..................................... 9
2.3 Blueprint of the designed bracket ............................................. 10
2.4 Acoustic signal recording using Kistler which is fixed at a close point to the physical contact on acrylic sheet ............................................. 11
2.5 MPLAB’s IDE, which is used to program e-Puck ............................ 13
2.6 TinyBootloader ............................................................... 13
2.7 Putty for the initial connection check ........................................ 14
2.8 A view from NI Signal Express with the data set of 11 taps .............. 15
2.9 A view from NI Signal Express with different data sets created from 11 tap signal ......................................................... 15

3.1 Longitudinal and shear waves by NDT Resource Center .................. 19
3.2 Acoustic emission wave created with tapping the accelerometer ...... 21
3.3 Zoom in of Figure 3.2 ...................................................... 21
3.4 Zoom out of Figure 3.2 ...................................................... 22
3.5 Zoom out of Figure 3.4 ...................................................... 22
3.6 Zoom out of Figure 3.5 ...................................................... 23
3.7 Acoustic signal recording using Kistler on e-Puck robot ................. 24
3.8 Acoustic signal recording using e-Puck .................................... 24

4.1 Triangulation technique by Kundu et al. .................................... 26

5.1 Random Variable X by Shin et al. .......................................... 30
5.2 System which has random input and random output ...................... 31
5.3 Sample time history of tossing a coin ..................................... 34
5.4 Example of an ensemble of tossing a coin .................................. 34

6.1 An illustration of the grid for LTM Method ................................ 40
6.2 A block diagram of LTM Method .......................................... 40
6.3 An illustration of CLTM Method in our setup ............................. 43
6.4 Signal recorded by Kistler accelerometer as a result of tapping by stepper motor ......................................................... 45

7.1 Pure delay on a zero-phase signal by Shin et al. ............................ 49
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Overview of Signal Grid and corresponding data labels</td>
<td>16</td>
</tr>
<tr>
<td>3.1</td>
<td>Overview of wave types in solids</td>
<td>20</td>
</tr>
<tr>
<td>6.1</td>
<td>Template overview for CLTM</td>
<td>43</td>
</tr>
<tr>
<td>6.2</td>
<td>Calculation of accuracy</td>
<td>46</td>
</tr>
<tr>
<td>6.3</td>
<td>Results of CLTM</td>
<td>46</td>
</tr>
<tr>
<td>7.1</td>
<td>Results of GLTM</td>
<td>55</td>
</tr>
<tr>
<td>8.1</td>
<td>Results of Infinite Impulse Response Filter (IIRF)-based LTM with 2nd order IIRF</td>
<td>62</td>
</tr>
<tr>
<td>8.2</td>
<td>Results of IIRF-based LTM with 3rd order IIRF</td>
<td>63</td>
</tr>
<tr>
<td>8.3</td>
<td>Results of IIRF-based LTM with 4th order IIRF</td>
<td>63</td>
</tr>
<tr>
<td>8.4</td>
<td>Results of Transfer Function-based Location Template Matching (LTM)</td>
<td>66</td>
</tr>
</tbody>
</table>
List of Acronyms

CLTM  Correlation-based Location Template Matching
FIR  Finite Impulse Response
FRF  Frequency Response Filter
GLTM  Group Delay-based Location Template Matching
IIR  Infinite Impulse Response
IIRF  Infinite Impulse Response Filter
LTM  Location Template Matching
TOA  Time of Arrival
TDOA  Time Difference of Arrival
Acknowledgments

I wish to express my deepest gratitude to those who has touched the process of this thesis work in any sense and made it come true. Especially, my adviser Rifat Sipahi, my friend Melda Ulusoys and other colleagues in Egan Research Center, who helped and guided me through this long and exhausting process.

I also would like to thank for the generous and continuous supports of Turkish Petroleum Corporation and my adviser Rifat Sipahi to the fullest extent.
Abstract of the Thesis

Location Template Matching-based Study of Acoustic Emission Localization using e-Puck Robots

by

Cagri Dikilitas

Master of Science in Mechanical Engineering

Northeastern University, August 2015

Acoustic source localization is investigated in this thesis using various methods. These methods as proposed in the literature are generally conducted in highly controlled environments and using state-of-art technologies. In this study, the focus is on a scenario where one does not have the best instrumentation tools but a low capacity mobile robot called e-Puck with relatively limited hardware capabilities. On this e-Puck robot system, acoustic emission localization techniques are studied using the built-in accelerometer and Bluetooth communication onboard the robot. For this, a working experimental setup is built with proper programming combinations and data conversions, and the efficacy of four different acoustic source localization techniques are evaluated through their implementation on the e-Puck platforms. These techniques are specifically based on the idea of template matching, in either time-domain or frequency-domain. Moreover, the software implementation is also validated with a Kistler accelerometer and the efficacy of the studied algorithms with respect to the conditions applied on the e-Puck robot is compared. In light of the experimental results, this thesis provides insights into acoustic emission localization with limited sensing capabilities of the e-Puck robots, evaluation of the efficacy of the studied techniques under less ideal conditions, and discussions on inherent technical limitations of hardware and software, as well as on the details of the experimental setup built and ideas for future work.
Chapter 1

Introduction

Estimation of an acoustic source whereabouts relative to a reference frame is known as acoustic source localization [1]. In this thesis, we will be using two different types sensors and four different location template-based methods to estimate the location of an acoustic emission source on a cast acrylic sheet.

1.1 Motivation for Research

Acoustic source localization is a very common technique which is used in numerous applications. It is possible to implement these techniques to a wide variety of applications from development of touch screens to structural health monitoring [2], mechanical failure detection, military applications (such as artillery detection) [3] and it is even possible to track ships [4] and convert solid surfaces into a touch pad [5].

This thesis aims to answer the following questions: Could e-Puck robots be used for acoustic emission purposes? What are the hardware and software needed for the acoustic emission localization implementation? How does such implementation on an e-Puck robot result in terms of performance when compared to the results in literature and other hardware available in laboratory?

1.2 Goal of Study

In this thesis, acoustic localization of a vibration source in two dimensions is studied. During the study, many different techniques for different array of sensors and configurations are
reviewed to find the most appropriate method for our current configuration of sensors and environment.

For a given setup, which consists of a two dimensional medium, an accelerometer and acoustic emission source, here we have estimated the position of emission source relative to the sensor location. For this, we have used simplistic approaches as much as possible that do not require knowledge of the mechanical properties of the medium and that are less expensive in terms of both computation and hardware.

The reason behind avoiding mechanical properties was, if we could utilize a technique that works without a mathematical model, it could be possible to implement these methods to many different structures regardless of their complexity. In this setting, it would be possible to deliver a high level localization without the need to deal with low level understanding of material properties and other structural constraints. With this perspective, it would also be possible to adjust the computation of algorithms easily, in case the properties and circumstances change.

Here, our goal is to successfully estimate a vibration source using single and/or multiple e-Puck robots using different techniques. It is also desired that the estimation has ability to learn and adapt to changes in the experimental setting.

Using mobile and relatively cheaper equipment could also be an advantage for extending the application area of these techniques. In order to accomplish this goal, several methods in literature that were applied using a small education purposed robot. Having this done, we aimed to develop a detailed comparison of the estimation methods on the current experimental setup.

### 1.3 Acoustic Source Localization Methods in Literature

There are many techniques that can be used depending on the environment, types of sensors and characteristics of acoustic emission source. Among this large range of methods, we distilled some of the most popular techniques to be used in this experiment.

Methods that we are going to explain, compare and use for localization are:

- Triangulation [6]
- Correlation-based Location Template Matching (CLTM) [5]
- Group Delay-based Location Template Matching (GLTM) [7]
- IIRF-based Location Template Matching [5]
CHAPTER 1. INTRODUCTION

1.4 Assumptions and Limitations

In this thesis, different test beds were used but an acrylic sheet with dimensions as 48” x 48” x 5/16” in the experimental setup details was decided as the resultant test bed. Two different types of sensors (Kistler 8688A10 Accelerometer and e-Puck robot) were used to detect acoustic emission source and the above listed algorithms were utilized to track the acceleration data.

Kistler accelerometer was used for comparison and validation purposes on the performance of the algorithms and the efficiency of these algorithms when implemented using the accelerometer on the e-Puck robot. That Kistler accelerometer was also used as a guide and a more reliable external help resource in times it was needed for algorithm checks.

Acoustic emission source was designed to be used in specific conditions as required by the methods. It was an isolated source, which produces acoustic emission by tapping onto an acrylic sheet at a specified constant frequency. Different from the existing studies, here the source is fixed - but not the accelerometers, and localization of the source must be achieved at certain grip points away from the source.

General view of the experimental setup with the stepper motor and e-Puck robots can be seen in Figure 1.1.

1.5 Steps of this Research

In order to conduct the outlined research we have gone through the following:

- We did not have a full knowledge of the capabilities and the limitations of the e-Puck robot. We have understood the hardware that could be used for acoustic emission localization purposes. Once we have discovered that e-Puck had an accelerometer on board, we built the proper working environment for the e-Puck.

- We have found out that MPLAB IDE v7.20 was being used to create .hex files which includes the program running on e-Puck. Once we were able to create the .hex files, we have used a program called TinyBootloader to send our code to e-Puck via Bluetooth.

- We have inspected the programming manual of e-Puck robots and several programming examples to understand the coding principles and came up with the code which constantly reads the accelerometer data from e-Puck and sends out the values using e-Puck’s Bluetooth.
CHAPTER 1. INTRODUCTION

Figure 1.1: General view of experimental setup
CHAPTER 1. INTRODUCTION

connection to the computer that we use. We have acquired data via the serial port on PC side, using the program named Putty.

- We have decided to utilize a more reliable second sensor to validate and compare algorithms. We have used Kistler’s accelerometer with the properties given in the following chapters.

- We have used LabVIEW to log the data from both Kistler and e-Puck. For Kistler accelerometer it was easier to implement thanks to Quanser’s DAQ card’s compatibility and easy to understand process. For e-Puck, we have created a LabVIEW code which connects to the serial port and logs the data transmitted from e-Puck according to our needs.

- We had to have a proper, controllable acoustic emission source. We have used a stepper motor and Arduino to control the motor. We have also designed and manufactured the appropriate bracket out of aluminum to fix the stepper motor and an arm out of polyester to create controlled excitation by tapping onto the acrylic sheet for the acoustic emission purposes.

- Once we had established those, we have prepared the localization algorithms and appropriate data conversion software for data compatibility using MATLAB.

- Next, we have conducted the experiments with the scenario of a fixed acoustic emission source. We have prepared algorithms in MATLAB environment, collected the data and analyzed it for both e-Puck robots and Kistler accelerometer.

- We finally provided comparisons, accuracy of the four algorithms utilized, and improvement for future work.
Chapter 2

Experimental Setup, Hardware and Software

In this chapter, details about the experimental setup and hardware are presented.

2.1 Mobile Robot e-Puck

e-Puck is a mobile, mini scale sized robot (see Figure 2.1 Source:[11]) which is developed for teaching purposes by Swiss Federal Institute of Technology in Lausanne (EPFL).

![Figure 2.1: e-Puck robot](image)
CHAPTER 2. EXPERIMENTAL SETUP, HARDWARE AND SOFTWARE

Compared to its small size, it has many different sensors and utilizes a dsPIC processor. Open source software and hardware provides low-level access and it is very useful when it is used for intentions of developing programming skills. It is common to see e-Puck in action as the main hardware for many computer science lectures [12].

Specifications of the e-Puck are given as the following [13];

- Dimensions - 70 mm diameter, 55 mm height, 150 g weight
- Microcontroller - dsPIC 30F6014A at 60MHz (15MIPS) 16 bit with DSP core for signal processing
- Memory - 8 KB RAM, 144 KB Flash Memory
- Movement capabilities - 2 Stepper motors which have 50:1 gear ratio and 0.13 mm resolution. Reaches 15 cm/s velocity.
- Light sensors - Ambient light and 8 infra-red proximity sensors
- 3D accelerometer
- 3 omni-directional microphones
- VGA color camera
- IR receiver for remote control
- 8 red led lights around the robot
- Speaker for WAV files
- RS232 and Bluetooth connection ports
- 5 Watt-hour Li-Ion battery which lasts about 2 hours
- 16 position rotating switch for different program usage
- GNU GCC compiler system which provides C programming

We have used e-Puck robots to test and illustrate capabilities and effectiveness of small and mobile devices for the localization problem at hand. Ultimate goal is to determine the best solution for our given acoustic localization problem using this e-Puck robot.
CHAPTER 2. EXPERIMENTAL SETUP, HARDWARE AND SOFTWARE

2.2 Kistler Accelerometer

A tri-axial accelerometer, Kistler Type 8688A10 is used to validate the efficiency of the four algorithms.

Kistler Type 8688A10 has the following properties [8]

- Range: ±10g
- Sensitivity: ±500mV/g
- Frequency response: 0.5 – 4000Hz
- Threshold: 0.00016grms
- Power supply voltage: 20 – 30VDC
- Power supply current: 2 – 20mA
- Housing Material: Titanium

2.3 Acoustic Emission Source

A NEMA-17 size stepper motor shown in Fig.2.2 [14] controlled by Arduino, with the properties listed below [15], is used to do tapping at a given point to generate a controlled acoustic emission.

- 200 steps per revolution, 1.8 degrees
- Coil 1: Red and Yellow wire pair. Coil 2 Green and Brown/Gray wire pair.
- Bipolar stepper which requires 2 full H-bridges
- 4-wire, 12 inch leads
- 42 mm/1.65” square body
- 31 mm/1.22” square mounting holes, 3 mm metric screws (M3)
- 5 mm diameter drive shaft, 24 mm long, with a machined flat
CHAPTER 2. EXPERIMENTAL SETUP, HARDWARE AND SOFTWARE

Figure 2.2: Adafruit’s NEMA-17 size stepper motor

- 12 V rated voltage (can be driven at a lower voltage, but the torque will drop) at 350 mA max current
- 28 oz-in, 20 N-cm, 2 kg-cm holding torque per phase
- 35 Ohms per winding

In order to have this stepper motor fixed appropriately and to provide an adequate repeated tapping effect, a mounting bracket is designed and machined out of aluminum, Figure 2.3.

This assembly allowed us to create tappings which generate acoustic emission to be used in our implementation. With the creation of multiple taps, the signal which is illustrated in Figure 2.4 is generated at the point of impact. These taps were recorded using Kistler accelerometer fixed onto the acrylic plate at a very close point of impact. A direct hit on the accelerometer is tried and the results are shown in Chapter 3.

Signals from the taps were intended to check consistency of the created acoustic emission signal. When we take the first tap and correlated each taps after the first tap using cross-correlation coefficient, we have found that the average correlation between each tap is 0.965. If we had two
CHAPTER 2. EXPERIMENTAL SETUP, HARDWARE AND SOFTWARE

Figure 2.3: Blueprint of the designed bracket
Figure 2.4: Acoustic signal recording using Kistler which is fixed at a close point to the physical contact on acrylic sheet
identical taps, correlation coefficient would be equal to 1 but with this given correlation results it is shown that taps are close in terms of consistency.

2.4 Emission Propagation Medium - Acrylic Sheet

After several tryouts with different mediums available in laboratory, we wanted to have a dedicated medium throughout the experiments. In order to do so, we have ordered an acrylic sheet with the dimensions of 48” x 48” x 5/16”. We have isolated that sheet using conical, polyurethane, adhesive-back bumpers with dimensions 13/16” wide and 7/8” thick. This idea was initially observed during literature research [5].

2.5 Communication and Software

In this setup, we have used a desktop computer which has a Quanser - NI E-Series Terminal Board which is compatible with Kistler accelerometer and a Kistler Type 5134B Interface/Coupler to amplify outputs and edit sensor settings. A serial port connection is established with the help of an additional Bluetooth-usb dongle to the same computer in order to be able to program and read outputs of e-Puck robot. We would like to mention that, while sending out the accelerometer data from the e-puck to desktop computer, we are aware that sampling period will not necessarily remain the same due to the serial communication capabilities of e-Puck. In order to overcome this inherent hardware difficulty, we have decided to make the assumption that sampling period is constant, which is found by dividing the total data collection time by the total sample counts collected. Nevertheless, with the same comparison tools that we used in our algorithms, we have minimized the side effects of this assumption.

MPLAB’s IDE is used to develop programs for e-Puck appropriately. We have used that IDE to make e-Puck send out accelerometer readings through Bluetooth constantly. TinyBootloader is used to embed hex files to e-Puck, which are previously created using MPLAB’s IDE. Moreover, Putty is used to make sure the initial connection is established and Bluetooth connection is ready to log accelerometer data.

LabVIEW and MATLAB are used to communicate with the sensors and log outputs accordingly. Both LabVIEW and MATLAB are also used for algorithm development initially. We also implemented the entire algorithm using MATLAB, due to easier and parametric editing capabilities and personal preference.
CHAPTER 2. EXPERIMENTAL SETUP, HARDWARE AND SOFTWARE

Figure 2.5: MPLAB’s IDE, which is used to program e-Puck

Figure 2.6: TinyBootloader
CHAPTER 2. EXPERIMENTAL SETUP, HARDWARE AND SOFTWARE

2.6 Data Preparation

After setting up the environment detailed above, next step is to log and prepare data sets. We have used proper data sets to create a template from 10 tap signals and have logged another set of data from each node to use as the unknown emission source. With these preparations, we had the signals with 11 taps logged as shown below.

We have logged 6 files with 11 taps in each per node. Once these signals are logged as explained above, we spared the last tap from each signal (as shown in blue) and saved as the reference signal to be used. We named that signal as $y_1$ for node 1, $y_2$ for node 2 and so on. The remaining 10 taps (as shown in lime) are named as $x_{11}, x_{12} ... x_{16}$. Lastly, randomly selected single tap is saved separately (as shown in pink) to be used in single tap results.

With all the data sets created, the grid can be shown as below in table 2.1.
CHAPTER 2. EXPERIMENTAL SETUP, HARDWARE AND SOFTWARE

Figure 2.8: A view from NI Signal Express with the data set of 11 taps

Figure 2.9: A view from NI Signal Express with different data sets created from 11 tap signal
### Table 2.1: Overview of Signal Grid and corresponding data labels

<table>
<thead>
<tr>
<th>Node</th>
<th>Reference Signal</th>
<th>Single Tap</th>
<th>10 Tap</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 (p7)</td>
<td>$y_{71}, y_{72}, \ldots, y_{76}$</td>
<td>$x_{71s}, x_{72s}, \ldots, x_{76s}$</td>
<td>$x_{71m}, x_{72m}, \ldots, x_{76m}$</td>
</tr>
<tr>
<td>8 (p8)</td>
<td>$y_{81}, y_{82}, \ldots, y_{86}$</td>
<td>$x_{81s}, x_{82s}, \ldots, x_{86s}$</td>
<td>$x_{81m}, x_{82m}, \ldots, x_{86m}$</td>
</tr>
<tr>
<td>9 (p9)</td>
<td>$y_{91}, y_{92}, \ldots, y_{96}$</td>
<td>$x_{91s}, x_{92s}, \ldots, x_{96s}$</td>
<td>$x_{91m}, x_{92m}, \ldots, x_{96m}$</td>
</tr>
<tr>
<td>4 (p4)</td>
<td>$y_{41}, y_{42}, \ldots, y_{46}$</td>
<td>$x_{41s}, x_{42s}, \ldots, x_{46s}$</td>
<td>$x_{41m}, x_{42m}, \ldots, x_{46m}$</td>
</tr>
<tr>
<td>5 (p5)</td>
<td>$y_{51}, y_{52}, \ldots, y_{56}$</td>
<td>$x_{51s}, x_{52s}, \ldots, x_{56s}$</td>
<td>$x_{51m}, x_{52m}, \ldots, x_{56m}$</td>
</tr>
<tr>
<td>6 (p6)</td>
<td>$y_{61}, y_{62}, \ldots, y_{66}$</td>
<td>$x_{61s}, x_{62s}, \ldots, x_{66s}$</td>
<td>$x_{61m}, x_{62m}, \ldots, x_{66m}$</td>
</tr>
<tr>
<td>1 (p1)</td>
<td>$y_{11}, y_{12}, \ldots, y_{16}$</td>
<td>$x_{11s}, x_{12s}, \ldots, x_{16s}$</td>
<td>$x_{11m}, x_{12m}, \ldots, x_{16m}$</td>
</tr>
<tr>
<td>2 (p2)</td>
<td>$y_{21}, y_{22}, \ldots, y_{26}$</td>
<td>$x_{21s}, x_{22s}, \ldots, x_{26s}$</td>
<td>$x_{21m}, x_{22m}, \ldots, x_{26m}$</td>
</tr>
<tr>
<td>3 (p3)</td>
<td>$y_{31}, y_{32}, \ldots, y_{36}$</td>
<td>$x_{31s}, x_{32s}, \ldots, x_{36s}$</td>
<td>$x_{31m}, x_{32m}, \ldots, x_{36m}$</td>
</tr>
</tbody>
</table>
Chapter 3

Wave Motion in Elastic Solids

Here, we would like to give a summary of the mechanisms which take place in wave motion. This information will be useful to generate a better understanding of the wave propagation and interpret its possible effects on measured acceleration signals.\footnote{This chapter is a summary gathered from the works of [16] and [17]. No copyright infringement is intended. Symbols and terms are followed as in the works of [16] and [17] to satisfy the compatibility with the original works of [16] and [17].}

3.1 Wave Propagation

Wave propagation is described as the transmission of the resultant of an applied disturbance to the other portions of an elastic medium. Examples of this phenomenon can be given as propagation of sound waves in the air, propagation of earthquakes through the earth or ripples of water as a result of previously thrown stone. This type of propagation can be observed in similar forms in elastic media, which can be solid, liquid or gaseous. Despite the many similarities in terms of background and physical effects, there are a lot of differences in wave propagation mechanisms as well. Due to these differences, here we will be focusing on the wave propagation in solid structures, which are resultant of mechanical disturbances and excitation. The roots of these described propagation is the result of physical influence of atoms within the structure with one another. Here we provide a brief overview of some of the key elements that we will be considering throughout our experiments.

Everything in the nature consist of tiny particles, but if we inspect the media as a continuous structure, we can simulate the medium by using mass, spring and damper representation.
CHAPTER 3. WAVE MOTION IN ELASTIC SOLIDS

this configuration, mass and stiffness parameters affect the propagation speed, where mass and elasticity are functions of mass density and elastic moduli. While these two functions characterize the system, interaction of the parts within the structure is characterized by a three-dimensional spring behavior. An outward wavefront is a good representation of the emission propagation. Within this motion, particles behind the front experience the wave motion and continue to oscillate for a while and particles ahead of the front do not experience the wave yet.

When we think of solid structures in wave motion, the properties which we have mentioned above result in two types of motion. First, transmission of tensile and compression stresses in the direction of the wave. This motion is similar to the mechanisms of wave propagation in fluids. Secondly, solids can transmit shear stress in addition to the firstly mentioned stresses. It should be noted that shear stress propagation will be transverse to the direction of the actual wave propagation.

With these in mind, it is useful to note that, outward propagation of the wave in a finite media will reach and interact with the boundaries of the structure. In a single wave propagation in a solid structure scenario, that wave will produce both compression and shear waves on an interacted boundary but acoustic emission will generate only its own type of waves. This continuous propagation and reflection behavior of waves bring the structure to the static equilibrium in the end, but the loading process itself is a dynamic process. This phenomenon must be considered critical when loading rates and transit times of the waves are comparable.

Wave propagation in solids can be divided into three general types; elastic waves, visco-elastic waves and plastic waves. Elastic waves are observed when Hooke’s law can be applied to explain behavior of material stress. If the behavior can be explained using both viscous and elastic stresses in act, then it is named as visco-elastic waves. For the case in which yield stress of the material is exceeded, plastic waves are of concern.

3.2 Types of Wave Propagation

We have mentioned earlier that, when a solid structure is under stress within the elastic limits, particles within that structure exhibit elastic oscillations. Right after the particles of the medium are being disturbed, they change their equilibrium position and internal forces arise. These internal forces, combined with the inertia of the particles lead to oscillations.

Wave propagation in solids is based on four principle modes: Longitudinal waves, shear waves, surface waves and plate waves (in thin plates). Longitudinal waves and shear waves are
CHAPTER 3. WAVE MOTION IN ELASTIC SOLIDS

the most commonly used and the particle movement by the effect of these waves are illustrated as shown below by NDT Resource Center [17].

Figure 3.1: Longitudinal and shear waves by NDT Resource Center (Source [17])

Longitudinal waves are also called as pressure or compression waves, due to the active compression and dilation forces. They are sometimes called density waves owing to the fluctuations in particle density as particles move. For the case of shear wave, particles oscillate perpendicular to the direction of propagation and they are weak in relation to the longitudinal waves. Due to the requirement of solid media, shear waves cannot propagate effectively through the liquids and gases.

There are other types of waves that can occur in solids due to the support of vibrations in other directions. Especially at surfaces and interfaces, different types of vibration are possibly generated. A table of wave types in solids are provided in the following table [17]:

Surface waves (Rayleigh) can occur in the surface of a thick solid and can penetrate to one wavelength depth. Both longitudinal and shear motion combined, and elliptic motion is achieved. Surface waves likely to occur when longitudinal wave intersects a surface near the second critical angle. Plate waves are similar to Rayleigh waves but they can occur only in materials with a few wavelengths thick. Lamb waves are a good example of the plate waves and they propagate parallel to the surface through the thickness of the medium. Lamb waves can travel several meters in steel[17].

---

2Second critical angle is defined by NDT Resource Center [18] as the angle of refraction for the perpendicular shear wave where all of the wave energy is reflected to the next point that follows the shear wave.
CHAPTER 3. WAVE MOTION IN ELASTIC SOLIDS

<table>
<thead>
<tr>
<th>Wave Types in Solids</th>
<th>Particle Vibrations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Longitudinal</td>
<td>Parallel to wave direction</td>
</tr>
<tr>
<td>Transverse (Shear)</td>
<td>Perpendicular to wave direction</td>
</tr>
<tr>
<td>Surface - Rayleigh</td>
<td>Elliptical orbit - symmetrical mode</td>
</tr>
<tr>
<td>Plate Wave - Lamb</td>
<td>Component perpendicular to surface (extensional wave)</td>
</tr>
<tr>
<td>Plate Wave - Love</td>
<td>Parallel to plane layer, perpendicular to wave direction</td>
</tr>
<tr>
<td>Stoneley (Leaky Rayleigh Waves)</td>
<td>Wave guided along interface</td>
</tr>
<tr>
<td>Sezawa</td>
<td>Anti-symmetric mode</td>
</tr>
</tbody>
</table>

Table 3.1: Overview of wave types in solids

3.3 Waves and Signals In Our Experiment

3.3.1 Waves and Signals with A Direct Hit On The Accelerometer

The signal generated by a direct hit by the stepper motor onto the Kistler Accelerometer is illustrated in the following figures to investigate the wave reflections from the boundaries of the acrylic sheet and their time of arrival.

Figure 3.2 shows the initial impact on the accelerometer and the reflection of the wave from three relatively near boundaries of the medium. Figure 3.3 shows a zoomed-in version of Figure 3.2 where the initial impact on the accelerometer and the reflection of the waves from the boundaries of the medium are clearly visible. Figure 3.4 shows the damping behavior of the medium by zooming out Figure 3.2.

Moreover, in Figure 3.5 and Figure 3.6 we see the reflection from the edges. It should be noted that the time difference between original signal and the reflections are 0.04, 0.067 and 0.076 seconds respectively. The detectable reflections end about 0.1 seconds. That’s why we will implement the algorithms with repeated signals generated in longer periods such as over 1.2 seconds, to account for reflection effects in the detection algorithms.

With the illustration of the Figures 3.2, 3.3, 3.4, 3.5, and 3.6 we showed the reflection behavior of the acoustic emission waves in the experimental setup that we have built. As seen on Figure 3.5 and 3.6, we have noticed another signal (∼4 seconds after tapping) received in the accelerometer when we tap directly onto the accelerometer. But this second signal does not occur at the times when we did the tapping not directly to the accelerometer but a point very close to the accelerometer location. We guess it is a natural behavior of the accelerometer itself, which exist due.
CHAPTER 3. WAVE MOTION IN ELASTIC SOLIDS

Figure 3.2: Acoustic emission wave created with tapping the accelerometer

Figure 3.3: Zoom in of Figure 3.2
CHAPTER 3. WAVE MOTION IN ELASTIC SOLIDS

Figure 3.4: Zoom out of Figure 3.2

Figure 3.5: Zoom out of Figure 3.4
3.3.2 Waves and Signals Acquired on e-Puck and Kistler

When we use Kistler on e-Puck robots, a sample signal created by the tapping behavior is as given in Figure 3.7. In the configuration of Kistler being placed on top of the e-Puck, reflections are barely visible. This is very likely because of added compliance with e-Puck robot.

Moreover, when we consider acceleration acquisition using e-Puck robots, a sample signal created by the tapping behavior is as given in Figure 3.8. With the additional damping affects of e-Puck robot and poor DAQ capabilities, we can only acquire the initial tap and reflection signals, if they exist, are barely noticeable.

Another point that we would like to mention here is that, even if there are wave reflections from the boundaries, they are position dependent. Since Location Template Matching algorithms use the same position for the signals, the reflection behavior of the medium is expected to be similar. Thus, the reflection behavior is not expected to affect the selected algorithms.
CHAPTER 3. WAVE MOTION IN ELASTIC SOLIDS

Figure 3.7: Acoustic signal recording using Kistler on e-Puck robot

Figure 3.8: Acoustic signal recording using e-Puck
Chapter 4

Triangulation

4.1 Overview

When a structure is stressed, it will generate acoustic emission and that acoustic emission will arrive to different sensors at different times. Triangulation technique is based on the time differences of wave arrivals at these sensors. That requires at least three measurement sensors and can be used for both known and unknown material properties [19].

Triangulation is one of the most popular, well-known and oldest source localization techniques which uses this phenomenon primarily for isotropic and homogeneous structures.

4.2 Details of Method

Consider an array of three sensors located at three points, denoted as $S_0$, $S_1$ and $S_2$. For such an array of three sensors, two hyperbolae will be defined through a pair of differences in arrival time. Intersection of these hyperbolae represents location of acoustic emission source.

As explained in [6], $P(x, y)$, acoustic emission source can be calculated by the intersection of three circles about $S_0$, $S_1$ and $S_2$. Radii are denoted by $r$, $r + \delta_1$ and $r + \delta_2$ and three circles are represented by:

$$
x^2 + y^2 = r^2$$
$$
(x - x_1)^2 + (y - y_1)^2 = (r + \delta_1)^2$$
$$
(x - x_2)^2 + (y - y_2)^2 = (r + \delta_2)^2
$$

(4.1)

and
CHAPTER 4. TRIANGULATION

Figure 4.1: Triangulation technique — three sensors placed at positions 1, 2 and 3 receive the acoustic waves generated by the source at position P. Radius of each circle corresponds to the distance traveled by the wave from the source to the sensor. (Source [19])

\[
\begin{align*}
\delta_1 &= t_1 \times \nu \\
\delta_2 &= t_2 \times \nu
\end{align*}
\]  

(4.2)

where \( \delta_1 \) and \( \delta_2 \) are distance differences and \( \nu \) is the propagation velocity of acoustic emission through the medium. Using the above equations, we obtain

\[
\begin{align*}
2xx_1 + 2yy_1 &= (x_1^2 + y_1^2 - \delta_1^2) - 2r\delta_1 \\
2xx_2 + 2yy_2 &= (x_2^2 + y_2^2 - \delta_2^2) - 2r\delta_2
\end{align*}
\]  

(4.3)

which, in polar coordinates, reads

\[
\begin{align*}
2r(x_1 \cos \theta + y_1 \sin \theta + \delta_1) &= (x_1^2 + y_1^2 - \delta_1^2) - 2r\delta_1 \\
2r(x_2 \cos \theta + y_2 \sin \theta + \delta_2) &= (x_2^2 + y_2^2 - \delta_2^2) - 2r\delta_2
\end{align*}
\]  

(4.4)

and since

\[
\begin{align*}
2r(x_1 \cos \theta + y_1 \sin \theta + \delta_1) \neq 0 \\
2r(x_2 \cos \theta + y_2 \sin \theta + \delta_2) \neq 0
\end{align*}
\]  

(4.5)
CHAPTER 4. TRIANGULATION

From Eq. 4.3

\[
\frac{r}{2r(x_1 \cos \theta + y_1 \sin \theta + \delta_1)} = \frac{(x_2^2 + y_2^2 - \delta_2^2) - 2r \delta_2}{2r(x_2 \cos \theta + y_2 \sin \theta + \delta_2)}
\]  

(4.6)

Denoting

\[ A_1 = x_1^2 + y_1^2 - \delta_1^2 \quad \text{and} \quad A_2 = x_2^2 + y_2^2 - \delta_2^2 \]  

(4.7)

Angle of the first sensor with respect to acoustic emission source can be found using the formula below;

\[
\tan \theta = \frac{(A_1 y_2 - A_2 y_1)}{(A_1 x_2 - A_2 x_1)}
\]  

(4.8)

The above summarized formulae would be used for known propagation speed, but a more generalized formula can be generated as shown in [19].

In terms of implementation, three sensors are attached to a plate. If the plate has an acoustic source at point P then the wave generated by the acoustic source propagates through the plate and strikes the sensors at different times. For plate type structure the generated wave is the Lamb wave [19]. Denoting times of travel of the wave to sensors 1, 2, 3 as \(t_1, t_2, t_3\) and exact clock time of the acoustic event \(T_0\). It is impossible to obtain travel times \(t_1, t_2, t_3\) from the times of detection (clock times) \(T_1, T_2, T_3\) of the arriving waves at the three sensors. However, it is possible to extract Time Difference of Arrival (TDOA);

\[
t_{ij} = t_i - t_j = T_i - T_j = T_{ij}
\]  

(4.9)

Even though travel times are unknown, it is still possible to find the distance traveled by acoustic waves reaching to sensor \(i\) and \(j\) as

\[
d_{ij} = t_{ij} \times \nu
\]  

(4.10)

Using Eq. 4.10 relation, for isotropic plate with unknown propagation speed, one can solve the following four nonlinear equations:

\[
\begin{align*}
    d_2 - d_1 &= d_{12} = t_{12} \times \nu \\
    d_3 - d_2 &= d_{23} = t_{23} \times \nu \\
    d_1 \sin \theta_1 &= d_2 \sin \theta_2 \\
    d_1 \cos \theta_1 + d_2 \cos \theta_2 &= (x_1 - x_2)^2 + (y_1 - y_2)^2 = (D_{12})^2
\end{align*}
\]  

(4.11)
CHAPTER 4. TRIANGULATION

where $D_{12}$ is the distance between sensors 1 and 2.

All these equations yield to six unknowns $\theta_1, \theta_2, d_1, d_2, d_3$ and $\nu$ but it is important to note that, they can be represented in terms of only three; $\nu, x_0$ and $y_0$. These are actually known for a given setup. This relation makes triangulation possible.

4.3 Various Experimental Examples and Accuracy

Triangulation has very wide areas of application since it relies on the basic principles and it is one of the very fundamental techniques that has shaped today’s knowledge. Some applications that uses Triangulation can be listed as:

- Detection of impact point [20]
- Determination of microphone location [21]
- Identification of impact force for smart composite stiffened panels [22]
- Acoustic gunshot detection and localization [3]
- Localization of birds e.g. Great Bitterns (*Botaurus stellaris*) with a maximum probabilistic radius of 10% [23]
- Speaker position detection [24]

4.4 Comments

As mentioned above, Triangulation is a very fundamental approach in acoustic emission localization. It has always been highly popular in this field of study but it has a relatively large draw-back, velocity of propagation of the acoustic emission has to be known. This is a property that can change easily when we deal with dynamical systems. This is one of the critical points for Triangulation that effects performance radically. It is often needed to increase number of sensors to improve accuracy. Yet it may not be enough, with the timing considerations taken into account.

This technique is a simple approach for acoustic source localization and it depends on precise calculation of arrival time and material properties, as well as the algorithms that have been developed. Since these parameters affect the result dramatically, it is often not that easy to apply
CHAPTER 4. TRIANGULATION

triangulation successfully and with high accuracy. Another handicap that triangulation has, is being vulnerable to noisy signals since it is harder to extract the Time of Arrival (TOA) accurately \[25\].

This method was used for known material properties but later it is shown that it is also possible to implement this method in the case when material properties are unknown. In such cases, due to the change in properties, the approach comes with some consequences, which make it more vulnerable to complexities, while sacrificing accuracy and resolution. For now, it is not possible to overcome this disadvantage. As a solution to this problem, there are alternative techniques to be preferred for better results (i.e. beamforming and optimization-based techniques) \[19\].

It is also worth noting that there are some hybrid techniques which combine beamforming and TDOA \[26\] but these will not be discussed in this thesis but can be considered for further studies.

Despite the fact that this method is one of the most fundamental approaches, it is a highly possible scenario that one can find themselves in a position that Triangulation is not the right solution for the specific problem due to the difficulties in implementation. We guessed we could experience that as we try to increase robustness of the setup yet we wanted to acknowledge this valuable method in this thesis. That’s how we wanted to show the reason why we aimed to use the other approaches better. We are going to mention details of them in the next chapters.

With these in mind, we wanted a system that does not require material properties. Plus it would be really hard to deal with the inner-clocks of each robot and synchronization of multiple robots, processors and computers. As a result, it would not be suitable and effective for our setup and we aimed for other algorithms which would satisfy our needs.
Chapter 5

Signal Processing and Statistics

Most of signal processing and the further algorithms that we are going to be using will require some statistics knowledge. Here, an overview is provided.\[1\]

5.1 Random Processes

5.1.1 Random Variables

Outcome of an experiment is defined as a set of numbers instead of actual elements which belong to sample space. In order to do so, a random variable is defined as a function on a sample space. For the example of coin tossing as given by Shin et. al. \[27\]; 1 represents head and 0 stands for tail. With these assumptions, function for the given coin tossing problem is represented as; \(X(H) = 1\) and \(X(T) = 0\) in particular but it could also be generalized as \(X(w_i)\) as for any element \(w_i\) in \(\Omega\)

\[\begin{align*}
\text{Sample Space} & \quad \Omega \\
H & \quad \Omega_x \\
T & \quad \Omega_x \\
\end{align*}\]

\[\begin{align*}
\text{Range Space} & \quad 1 \\
0 & \quad \Omega_x \\
\end{align*}\]

\[\begin{align*}
\text{Sample Space} & \quad (H, H) \\
(H, T) & \quad (T, H) \\
(T, T) & \quad \Omega \\
\end{align*}\]

\[\begin{align*}
\text{Range Space} & \quad 0 \\
1 & \quad 2 \\
\Omega_x & \quad \Omega_x \\
\end{align*}\]

Figure 5.1: Representation of Random Variable X (Source \[27\])

\[1\]This chapter is a summary which is gathered from the works of \[27\], \[28\] and \[29\]. No copyright infringement is intended. Symbols and terms are followed as in the work of \[27\] to satisfy the compatibility with the original work of \[27\].
CHAPTER 5. SIGNAL PROCESSING AND STATISTICS

Two types of random variables exist; discrete random variable and continuous random variable. Discrete random variable is when sample space consists of discrete number of elements. Similarly for uncountable values in sample space, it will be named as continuous random variable.

5.1.2 Discrete Random Variables

Due to the discretization of signals when we use DSPs, we will be focusing on discrete random variables in our experiments. Another example for a discrete random variable is rolling a dice. In rolling a fair dice example, random variable \( X = x_i \) characterizes the probability distribution as \( P[X = x_i] \) for \( x_i = x_1, x_2, \ldots \) satisfies that sum of all probabilities is equal to 1.

\[
\sum P[X = x_i] = 1 \tag{5.1}
\]

Since there are six surfaces on a dice, probability distribution function for each surface is equal to \( \frac{1}{6} \).

Shin et. al. have also mentioned that averages are used frequently to describe properties of random variables [27]. If we take a discrete random variable \( X \) as an example which has values \( x_1, x_2, \ldots \) and \( p_1, p_2, \ldots \) as probabilities. Let \( x_i \) happen \( n_i \) times in \( N \) trials. This leads to average value of \( X \) as;

\[
\bar{x} = \frac{1}{N} \sum_{i} n_i x_i \tag{5.2}
\]

where \( \bar{x} \) is called the sample mean. This leads to expected value (mean of probability distribution).

Theoretical mean value (or the first moment) of random variable \( X \) is described as \( E[X] \), the expected value of \( X \).

\[
E[X] = \int_{-\infty}^{\infty} x g(x) \, dx \tag{5.3}
\]

It is applied to discrete probability distribution functions as;

\[
E[X] = x_1 p_1 + x_2 p_2 + \ldots + x_k p_k \tag{5.4}
\]

This concept can be demonstrated better with the help of schema shown below;

![Figure 5.2: System which has random input and random output](image)
CHAPTER 5. SIGNAL PROCESSING AND STATISTICS

\[ E[g(X)] = E[Y] = \int_{-\infty}^{\infty} g(x)p(x) \, dx \]  
(5.5)

and for discrete process this formula is given by

\[ E[g(X)] = \sum_i g(x_i)p_i \]  
(5.6)

Of course, this formula is valid for only one random process. If we would like to expand this for cases which involves two processes, it becomes:

\[ E[W] = E[g(X,Y)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x,y)p(x,y) \, dx \, dy \]  
(5.7)

Even though characteristics of X, in terms of probability is contained in the probability density function \( p(x) \) it is generally handy to gather this information by using fewer parameters. This is achieved by utilizing moments of random variables. In general it is defined as;

\[ \mu'_n = E[X^n] = \int_{-\infty}^{\infty} x^n \, dF(x) \]  
(5.8)

First moment (or mean):

\[ \mu_x = E[X] = \int_{-\infty}^{\infty} x \, p(x) \, dx \]  
(5.9)

and second moment (or mean square):

\[ E[X^2] = \int_{-\infty}^{\infty} x^2 \, p(x) \, dx \]  
(5.10)

Moreover, variance (second moment about mean) is often used. It is defined as;

\[ Var(x) = (\sigma_x)^2 = E[(X - \mu_x)^2] = \int_{-\infty}^{\infty} (x - \mu_x)^2 \, p(x) \, dx \]  
(5.11)

where \( \sigma_x \) is the standard deviation. While mean, \( \sigma_x \), represents the location of probability density function \( p(x) \) on x-axis, variance represents dispersion of \( p(x) \) relative to \( \sigma_x \).

There are third and fourth moments as well but they are used for non-Gaussian processes. Furthermore, it would also be appropriate to mention, when \( \mu_x = 0 \) and \( \sigma_x^2 = 1 \) the distribution is called standard normal distribution.

The above formulae are valid for continuous signals. However will be dealing with digital signals. Thus, we should generate similar formulae for the use of digital signals within a calculable margin of error. The following formulae will be used for digital signals:
CHAPTER 5. SIGNAL PROCESSING AND STATISTICS

For a data set \((x_1, x_2, x_3, \ldots, x_N)\), which is acquired from \(N\) different measurements of a random variable \(X\), arithmetic mean \(\sigma_x\) will be calculated and named as the sample mean, \(\bar{x}\).

\[
\bar{x} = \frac{1}{N} \sum_{n=1}^{N} x_n
\]  
(5.12)

Variance \(\sigma_x^2\) can be estimated using the following formula and will be denoted as sample variance \(s_x^2\)

\[
s_x^2 = \frac{1}{N} \sum_{n=1}^{N} (x_n - \bar{x})^2
\]  
(5.13)

This equation gives a biased variance and often times it underestimates. Unbiased sample variance can be calculated using;

\[
s_x^2 = \frac{1}{N-1} \sum_{n=1}^{N} (x_n - \bar{x})^2
\]  
(5.14)

where \(s_x\) is the sample standard deviation,

\[
s_x = \sqrt{\frac{1}{N-1} \sum_{n=1}^{N} (x_n - \bar{x})^2}.\]

With the use of these formulae, unbiased sample covariance is estimated through;

\[
s_{xy} = \frac{1}{N-1} \sum_{n=1}^{N} (x_n - \bar{x})(y_n - \bar{y})
\]  
(5.16)

5.2 Stochastic Processes

5.2.1 Time-Dependent Random Variables

Random processes are time-independent. This property creates a division of usage for random variables and arises a similar but time involving phenomenon called stochastic processes.

If we consider the example of coin tossing, with time-dependency this time, time history could look like as depicted in Figure 5.3.

In the case of tossing a coin as a random process, range space \(X\) could be defined as \((1, 0)\). \(X\) is a random variable which is defined on a sample space heads or tail, \((H, T)\). Now we consider
as a stochastic process by introducing time and note $X(t)$ as a random function of time (or random variable indexed by time and defined on a sample space).

Figure 5.3 only a single try-out of stochastic process $X(t)$ is shown. It is not that hard to think that there could be different try-outs as indicated below. If we were to build a set of such try-outs are denoted by $X(t)$ and are called an ensemble. It may be important to remind that $t$ could be anything between $-\infty$ and $\infty$.

In order to define the distribution function at time $t$, let’s consider the probability density
function for a stochastic process, where $x$ is a particular value of $X(t)$;

$$F(x, t) = P[X(t) \leq x]$$

$$P[x < X(t) \leq x + \delta x] = F(x + \delta x, t)F(x, t)$$

which makes it possible to generate probability density function for a time-dependent stochastic process as;

$$p(x, t) = \frac{dF(x, t)}{dx}$$

With probability density function defined for time-dependent stochastic process, we can discuss moments of a stochastic process similar to random processes. First moment (or mean):

$$\mu_x(t) = E[X(t)] = \int_{-\infty}^{\infty} x p(x, t) dx$$

and second moment (or mean square):

$$E[X^2(t)] = \int_{-\infty}^{\infty} x^2 p(x, t) dx$$

also, we have that:

$$Var(X(t)) = (\sigma_x(t))^2 = E[(X(t) - \mu_x(t))^2] = \int_{-\infty}^{\infty} (x - \mu_x(t))^2 p(x, t) dx$$

5.2.2 Correlation Functions

5.2.2.1 Auto-correlation Functions

Auto-correlation function (or auto-covariance function) measures degree of association of one signal at time $t_1$ with the same signal, itself, at time $t_2$. Auto-correlation function is given as;

$$R_{xx}(t_1, t_2) = E[X(t_1)X(t_2)]$$

and if we were to subtract mean values, it would be named as auto-covariance function which is given as:

$$C_{xx}(t_1, t_2) = E[(X(t_1) - \mu_x(t_1))(X(t_2) - \mu_x(t_2))]$$
CHAPTER 5. SIGNAL PROCESSING AND STATISTICS

If we focus on statistical processes, it will be observed that statistical properties remain the same with shift of time. It would be possible to define Eq.(5.24) as a function of time difference only \((t_1 - t_2)\). Using this property, Eq.(5.24) would be simplified as;

\[
C_{xx}(t_1 - t_2) = E[(X(t_1) - \mu_x)(X(t_2) - \mu_x)]
\]  

(5.25)

If we note \(t_1 = t\) and \(t_2 = t + \tau\);

\[
C_{xx}(\tau) = E[(X(t) - \mu_x)(X(t + \tau) - \mu_x)]
\]  

(5.26)

Here, \(\tau\) is described as time lag. For the case of \(\tau = 0\);

\[
C_{xx}(0) = E[(X(t) - \mu_x)(X(t) - \mu_x)] = Var(X(t)) = (\sigma_x(t))^2
\]  

(5.27)

5.2.2.2 Cross-correlation Function

For two stochastic simultaneous processes \(X(t)\) and \(Y(t)\) (could be input-output), measure of association between \(X(t)\) at time \(t_1\) and \(Y(t)\) at time \(t_2\) can be given with cross-covariance function;

\[
C_{xy}(t_1 - t_2) = E[(X(t_1) - \mu_x)(Y(t_2) - \mu_y)]
\]  

(5.28)

or, when mean values are not subtracted, cross-correlation function;

\[
R_{xy}(t_1 - t_2) = E[X(t_1)Y(t_2)]
\]  

(5.29)

If we define these as a function of time difference, \(t_1 = t\) and \(t_2 = t + \tau\), similar to the notation used above, cross-covariance function could be re-written as the following;

\[
C_{xy}(\tau) = E[(X(t) - \mu_x)(Y(t + \tau) - \mu_y)]
\]  

(5.30)

and cross-correlation function as given in Eq(5.31);

\[
R_{xy}(\tau) = E[X(t)(Y(t + \tau)]
\]  

(5.31)

finally cross-covariance function is also expressed in terms of time lag as;

\[
C_{xy}(\tau) = R_{xy}(\tau) - \mu_x\mu_y
\]  

(5.32)
it is also worth noting that similar to cross-covariance function, cross-correlation function in terms of the time lag is expressed as:

\[ R_{xy}(\tau) = R_{yx}(-\tau) \]  (5.33)

In general, properties of covariance functions can be listed as;

1. auto-covariance function in normalized form (or non-dimensional) is;

\[ \rho_{xx}(\tau) = \frac{C_{xx}(\tau)}{\sigma_x^2} \]  (5.34)

and auto-correlation function (for zero mean) is;

\[ \rho_{xx}(\tau) = \frac{R_{xx}(\tau)}{R_{xx}(0)} \]  (5.35)

It is worth noting that autocovariance function also has the following properties:

(a) Autocovariance function is an even function;

\[ C_{xx}(\tau) = C_{xx}(-\tau) \]  (5.36)

and

\[ \rho_{xx}(0) = 1; \quad C_{xx}(0) = \sigma_x^2 \]  (5.37)

(b)

\[ |C_{xx}(\tau)| \leq \sigma_x^2; \quad |R_{xx}(\tau)| \leq R_{xx}(0) \]  (5.38)

Therefore;

\[ -1 \leq \rho_{xx}(\tau) \leq 1 \]  (5.39)

2. Cross-covariance function is defined as;

\[ \rho_{xy}(\tau) = \frac{C_{xy}(\tau)}{\sigma_x \sigma_y} \]  (5.40)

and cross-correlation function (for zero mean) is;

\[ \rho_{xy}(\tau) = \frac{R_{xy}(\tau)}{\sqrt{R_{xx}(0)R_{yy}(0)}} \]  (5.41)
(a) Cross-covariance function is neither even nor odd;

\[ C_{xy}(-\tau) = C_{yx}(\tau) \]  \hspace{1cm} (5.42)

and

\[ |C_{xy}(\tau)|^2 \leq \sigma_x^2 \sigma_y^2; \quad |R_{xy}(\tau)|^2 \leq R_{xx}(0)R_{yy}(0) \]  \hspace{1cm} (5.43)

Therefore,

\[ -1 \leq \rho_{xy}(\tau) \leq 1 \]  \hspace{1cm} (5.44)

(b) In case that time-domain signals \( X(t) \) and \( Y(t) \) are uncorrelated;

\[ C_{xy}(\tau) = 0; \quad R_{xy}(\tau) = \mu_x \mu_y \]  \hspace{1cm} (5.45)

These concepts are all summarized based on the work of Shin et. al. [27] so that we can use them in the algorithms which are detailed in the next chapters.
Chapter 6

Correlation-based Location Template Matching

6.1 Overview

Correlation-based Location Template Matching (CLTM) method requires only one measurement sensor and uses correlation coefficient to measure the rate of similarity between measured and the reference signals [7] without having the need to know dynamic properties of the medium.

Although CLTM method uses time signals and because of that property, its performance is highly affected by many factors (i.e. spatial resolution of reference points, frequency bandwidth of signal and length of data) it has extremely wide application areas and can be a competent algorithm for our goal.

6.2 Details of CLTM Method

One should generate an understanding of Location Template Matching (LTM) to digest LTM based techniques and algorithms. Location Template Matching is a technique that uses previously built signal template to estimate the location of newly acquired signal. First, a set of data is stored from previously determined points as a designated template and different properties of new signal is compared with the ones already available in the template. That procedure is called Location Template Matching since we are trying to match and find possible relations of the new data with the template. It would be appropriate to think of it as a look-up table.
CHAPTER 6. CORRELATION-BASED LOCATION TEMPLATE MATCHING

Depending on the property that we are going to use for determining relationship, it would be named differently (i.e. CLTM, GLTM or IIRF-based) but the fundamentals are the same.

In this chapter we explain the details of Correlation-based LTM. As mentioned earlier, we start with generating a template for whatever the conditions of the setup is. It could be a simple, perfectly homogeneous plate or a much more complex structure.

Here, if we consider the scheme depicted in Figure 6.1:

For each impact at the fixed location on the acrylic table, the vibration signals measured at the grip points $p_1...p_9$ through accelerometer is stored in a database and called as the reference signals, $x_1(t), x_2(t), ..., x_n(t)$ and $y(t)$. This database is used as the location template.

Figure 6.1: An illustration of the grid for LTM Method. (Source [7])

Figure 6.2: A block diagram of LTM Method. (Source [7])
CHAPTER 6. CORRELATION-BASED LOCATION TEMPLATE MATCHING

Later, if an arbitrary impact occurs near one of these predefined points, the newly measured vibration signal, \( y(t) \) is compared with all the signals in the database. Then, the location of impact source can be determined by finding the best matching signal in the database.

When we try to understand or emphasize any physical phenomenon, it is appropriate to have the complete mathematical representation. But once we fully understand the mathematical expression, we can find some characteristics that is unique and use them instead of the whole mathematical expression.

For CLTM method, we will use the cross-correlation coefficient. Let us present the details of the method and explain this concept.

Statistical Concepts:

Core structure of some concepts were given in the previous chapter so that we may continue in this chapter.

Returning to Eq. (5.7)

\[
E[W] = E[g(X, Y)] = E[XY] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} xy p(x, y) \, dx \, dy
\]

This relation is known as correlation between \( X \) and \( Y \). If we subtract mean values from each process for the sake of centralization, we have covariance between \( X \) and \( Y \).

\[
\sigma_{xy} = Cov(X, Y) = E[(X - \mu_x)(Y - \mu_y)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \mu_x)(y - \mu_y) p(x, y) \, dx \, dy
\]

which can also be written in form of

\[
Cov(X, Y) = E[XY] = \sigma_x - \sigma_y = E[XY] - E[X]E[Y]
\]

For different pairs of random variables \( X \) and \( Y \):

- if they are uncorrelated, \( Cov(X, Y) = 0 \) and \( E[XY] = E[X]E[Y] \)
- if they are orthogonal, \( Cov(X, Y) = -E[X]E[Y] \) and \( E[XY] = 0 \)
- if they are statistically independent, \( p(x, y) = p(x)p(y) \)

After the detailed explanation of these concepts, we can express a measure of linear relationship with utilizing correlation coefficient which is defined as:

\[
\rho_{x,y} = \frac{E[(X - \mu_x)(Y - \mu_y)]}{\sigma_x \sigma_y}
\]
CHAPTER 6. CORRELATION-BASED LOCATION TEMPLATE MATCHING

or

\[ \rho_{x,y} = \frac{\text{Cov}(X,Y)}{\sigma_x \sigma_y} \] (6.5)

Correlation coefficient has a value range between \(-1\) and \(1\), \(-1 \leq \rho_{x,y} \leq 1\), and reflects the degree of linear relationship between two random variables. Larger value for absolute value of \(\rho_{x,y}\) means better linear relation between \(X\) and \(Y\). CLTM algorithm is based on comparison of this linear relationship using time-domain signals. Correlation coefficient between newly acquired and previously recorded location template signal, is calculated and compared. CLTM algorithm seeks for the highest correlation coefficient to find the best matching point in the template. The point which can score the highest correlation coefficient with the signal from unknown location is declared as in the same spot.

Some of the application areas of CLTM can be listed as:

- Acoustic source localization [19], [7], [30], [31], [1]
- Acoustic ship-signature detection [4]
- Source location detection on smart composite stiffened panels [22] and thin plates [32]
- Touch interface development [33], [34]
- Gas pipeline leakage detection [35]
- Scene recognition and feature extraction [36]
- Video camera steering [37]
- Gunshot detection and localization [3]
- Automatic detection of microphone coordinates [38]

Finally, we note that improving the accuracy of CLTM technique requires more sensors to be used or larger space between points in the template.

6.3 Algorithm Implementation

Steps to develop CLTM algorithm are summarized next:
1. Define a grid to implement CLTM as illustrated in Figure 6.1
2. Collect signals from different points in the grid
3. Build a location template using the time domain signals collected from the grid
4. Log the time domain signal of which we want to estimate the position, from any point on the grid

![Figure 6.3: An illustration of CLTM Method in our setup](image)

5. Calculate cross-correlation coefficients one-by-one, using the newly acquired signal and the ones in the template

<table>
<thead>
<tr>
<th>Sensor Position</th>
<th>Time-domain Signal</th>
<th>Cross-Correlation Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>node 1 ((p_1))</td>
<td>(x_{11}, x_{12} \ldots x_{16})</td>
<td>(c_{11}, c_{12} \ldots c_{16})</td>
</tr>
<tr>
<td>node 2 ((p_2))</td>
<td>(x_{21}, x_{22} \ldots x_{26})</td>
<td>(c_{21}, c_{22} \ldots c_{26})</td>
</tr>
<tr>
<td>node 3 ((p_3))</td>
<td>(x_{31}, x_{32} \ldots x_{36})</td>
<td>(c_{31}, c_{32} \ldots c_{36})</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>node 9 ((p_9))</td>
<td>(x_{91}, x_{92} \ldots x_{96})</td>
<td>(c_{91}, c_{92} \ldots c_{96})</td>
</tr>
</tbody>
</table>

Table 6.1: Template overview for CLTM

6. Rearrange the grid with the calculated cross-correlation coefficient values
CHAPTER 6. CORRELATION-BASED LOCATION TEMPLATE MATCHING

7. Compare cross-correlation coefficient values and determine the signal pair which results in the highest cross-correlation coefficient value

8. Declare the signal pair with highest cross-correlation value as the source of the acoustic emission

With above given table 6.1, we can summarize the process as follows: Once we have 54 signals logged (9 nodes x 6 nodes /node), we calculate cross-correlation coefficients between the reference signal $y$ and each of the time-domain signals $x_{11}, x_{12}, ...$ to have 54 cross-correlation coefficients $c_{11}, c_{12}, ...$. Signal pair with the highest cross-correlation coefficient is declared as the output of CLTM algorithm. The node with which the signal from template is affiliated is declared to be the source of acoustic emission. For example; for the case of $x_{11}$ and $y$ has the highest cross-correlation coefficient, node 1 ($p_1$) is declared as the emission source.

6.4 Signals on e-Puck and Kistler

With the considerations on Chapter 3, we wanted to see how the signal is affected by the different configurations of the sensor. We have found that the signal recorded by the Kistler, with the tapping by the stepper motor resulted as seen in Figure 6.4.

6.5 Results and Comments

3x3 grid was used with 3 cm horizontal and 3 cm vertical distances between nodes. 6 sets of data were collected from each node and the same data collecting procedure was followed for both the single and 10 tap configurations. Table 6.3 was built with the collection of 108 data sets (54 for single tap and 54 for 10 taps) collected in total. Accuracy indicates on average across 9 nodes the correctly estimated node score of the algorithm, e.g. 50 correct guesses out of 54 is shown as 0.926. This approach is also given in Table 6.2.

Accuracy of CLTM is highly related to any disparity between signals. Even though more accurate results are reported in cm resolution in several other studies [33] and our experiments with Kistler accelerometer, in our experience this resolution could not be reached when we used the e-Puck robots. This may have a complex structure of different factors. When stepper motor’s low precision, poor isolation, e-Puck’s poor Bluetooth DAQ capabilities and accelerometer come together, it is highly probable to end up in high noise levels and also the reference acoustic emission
may not be repeated accurately. These all should have been the reason for this relatively poor accuracy. Nevertheless, we note that CLTM technique may also have some inherent limitations, since the average accuracy with the Kistler accelerometer was not substantially higher that that with the e-Puck robot.

Figure 6.4: Signal recorded by Kistler accelerometer as a result of tapping by stepper motor
CHAPTER 6. CORRELATION-BASED LOCATION TEMPLATE MATCHING

<table>
<thead>
<tr>
<th>Input Signal</th>
<th>Output Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y_1$</td>
<td>Correct $= 1$</td>
</tr>
<tr>
<td>$y_2$</td>
<td>Correct $= 1$</td>
</tr>
<tr>
<td>$y_3$</td>
<td>Wrong $= 0$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>$y_{54}$</td>
<td>Correct $= 1$</td>
</tr>
</tbody>
</table>

Total Accuracy $\frac{50}{54} = 0.926$

Table 6.2: Calculation of accuracy

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Excitation</th>
<th>Accuracy</th>
<th>Distance</th>
<th>Grid Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kistler</td>
<td>Single Tap</td>
<td>0.685</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
<tr>
<td>e-Puck</td>
<td>Single Tap</td>
<td>0.611</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
<tr>
<td>Kistler</td>
<td>10 Taps</td>
<td>0.666</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
<tr>
<td>e-Puck</td>
<td>10 Taps</td>
<td>0.574</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
</tbody>
</table>

Table 6.3: Results of CLTM
Chapter 7

Group Delay-based Location Template Matching

7.1 Overview

Another approach we would like to mention, investigate and implement for our localization goal, is called Group Delay-based Location Template Matching (GLTM). GLTM method requires only one measurement sensor and is used to locate acoustic emission source, similar to Correlation-based Location Template Matching (CLTM), without having to know the dynamic properties of the medium.

7.2 Details of GLTM Method

With the LTM concept in mind, as we change the parameter we use to compare template signals and newly acquired signal, we end up with a technique which has better accuracy.

In this sense, we will be changing the comparison tool to achieve GLTM. The main difference between CLTM and GLTM is that, GLTM uses frequency domain signal and group-delay property of the signal where CLTM uses correlation coefficient to measure the rate of similarity between measured and the reference signals in time domain [7].
CHAPTER 7. GROUP DELAY-BASED LOCATION TEMPLATE MATCHING

7.2.1 Theoretical Definitions

Phase delay is defined as the negative ratio of the angular frequency over frequency, given by;

\[
\tau_{\text{phasedelay}} = -\frac{\varphi}{\omega} = \frac{\theta}{f.360}
\]

(7.1)

Group delay is the negative slope (or derivative) of phase response with respect to the frequency. This property is the representation of the relative delay at different frequencies between input and output of a system.

\[
\tau_{xy} = -\frac{d\varphi}{d\omega} = \frac{d\theta}{df.360}
\]

(7.2)

where \( \varphi \) and \( \omega \) represent phase in radians and degrees, \( \theta \) and \( f \) are the angular frequency \([\text{rad/s}]\) and frequency \([\text{Hz}]\).

In general, group delay can be calculated as given below;

\[
\text{GroupDelay}(n) = -\left[ \frac{(\text{Phase}(n) - \text{Phase}(n - 1))}{f(n) - f(n - 1)} + \frac{(\text{Phase}(n + 1) - \text{Phase}(n))}{f(n + 1) - f(n)} \right] / 720
\]

(7.3)

7.2.1.1 Phase, Definition of Delay and Group-Delay

Even though we need both magnitude spectral density, \(|X(f)|\), and phase spectral density, \(\arg X(f) = \phi(f)\), to reconstruct a signal, often times phase spectral density is 'ignored' and only magnitude spectral density is drawn. On the other hand, phase spectral density is the key element that differs many different-looking signals with the same magnitude spectral density. In this manner, a linear phase change occurs when a pure delay is imposed on a signal.

Slope of the phase curve, gives the delay

\[
d\phi/df = -2\pi t_0
\]

(7.4)

or

\[
d\phi/dw = -t_0
\]

(7.5)

Here, \( t_0 \) is known as group delay of signal and since there is no dispersion, because of the pure delay the delay is the same for all frequencies.
CHAPTER 7. GROUP DELAY-BASED LOCATION TEMPLATE MATCHING

7.2.1.2 Continuous-time Linear Time-invariant Systems

Linear system is described as a system which satisfies the following two properties:

- **Additivity**

  ![Additivity property of a linear system](#)

- **Homogeneity**

  ![Homogeneity property of a linear system](#)

where \( x_1(t) \) , \( x_2(t) \) are inputs and \( y_1(t) \) , \( y_2(t) \) are responses of system.

Time-invariant system is described as a system whose time shift of input is responded with the same amount of time shift. This is illustrated by [27] as in the following figure;
CHAPTER 7. GROUP DELAY-BASED LOCATION TEMPLATE MATCHING

![Time Invariant System](image)

Figure 7.4: Illustration of time-invariant system (Source [27])

7.2.1.3 Impulse Response of a System and Frequency Response Function

Response of a linear systems to a unit impulse $\delta(t)$ at $t = 0$ is defined as impulse response of a system, $h(t)$, and illustrated as shown below:

$$x(t) = \delta t \quad \text{LTI System} \quad y(t) = h(t)$$

![Impulse Response of a System](image)

Figure 7.5: Illustration of impulse response of a system (Source [27])

For a linear time-invariant system, response to a delayed impulse $\delta(t - t_1)$ would be delayed impulse response $h(t - t_1)$ due to linearity. In the case of discretized, elemental, version of an arbitrary input signal $x(t)$ with $\Delta t_1$ intervals, impulse at time $t_1$ is $x(t_1)\Delta t_1$. It is also calculated that, due to the linearity of the system, response to this impulse at time $t$ is given as $h(t - t_1)x(t_1)\Delta t_1$. Addition of all the responses gives the total response of $y(t)$ at time $t$ as the following:

$$y(t) \approx \sum h(t - t_1)x(t_1)\Delta t_1$$

(7.6)

Considering a casual system, $h(t) = 0$ for $t < 0$, and letting $\Delta t_1 \to 0$, we have:

$$y(t) = \int_{-\infty}^{t} h(t - t_1)x(t_1) \, dt_1$$

(7.7)

and it is also worth noting that with the conversion $t - t_1 = \tau$, one obtains;
CHAPTER 7. GROUP DELAY-BASED LOCATION TEMPLATE MATCHING

Figure 7.6: Illustration of response of a system to elemental inputs (Source [27])

\[ y(t) = \int_{-\infty}^{t} h(t - t_1)x(t_1)\,dt_1 = -\int_{0}^{\infty} h(\tau)x(t - \tau)\,d\tau = \int_{0}^{\infty} h(\tau)x(t - \tau)\,d\tau \quad (7.8) \]

Figure 7.7: Illustration of impulse response function \( h(\tau) \) as weighting function or 'memory' (Source [27])

which is the well-known convolution integral [39]. With these definitions, here we summarize the given formulae as the following:

\[ y(t) = \int_{0}^{t} h(\tau)x(t - \tau)\,d\tau = \int_{0}^{t} h(t - \tau)x(\tau)\,d\tau \quad (7.9) \]

which is valid for a response if the input \( x(t) = 0 \) for \( t < 0 \) or 'casual' and

\[ y(t) = \int_{-\infty}^{\infty} h(\tau)x(t - \tau)\,d\tau = \int_{-\infty}^{\infty} (t - \tau)x(\tau)\,d\tau \quad (7.10) \]
which is valid for a response if the input \( x(t) \neq 0 \) for \( t < 0 \) or 'non-casual'. In this case, system also responds to future inputs.

**Frequency Response Function:**

Steady state response to a harmonic input such as \( x(t) = e^{j2\pi ft} \) is given as the following;

\[
y(t) = \int_{0}^{\infty} h(\tau)x(t - \tau) \, d\tau = \int_{0}^{\infty} h(\tau)e^{j2\pi f(t-\tau)} \, d\tau = e^{j2\pi ft} \int_{0}^{\infty} h(\tau)e^{-j2\pi f \tau} \, d\tau \tag{7.11}
\]

where frequency response function is given as the following;

\[
H(f) = \int_{0}^{\infty} h(\tau)e^{-j2\pi f \tau} \, d\tau \tag{7.12}
\]

for the response;

\[
y(t) = \int_{0}^{\infty} h(\tau)x(t - \tau) \, d\tau \tag{7.13}
\]

the Fourier transform of the response \( y(t) \) is;

\[
Y(f) = \int_{-\infty}^{\infty} \int_{0}^{\infty} h(\tau)x(t - \tau)e^{-j2\pi ft} \, d\tau \, dt \tag{7.14}
\]

Thus,

\[
Y(f) = H(f)X(f) \tag{7.15}
\]

and

\[
H(f) = Y(f)/X(f) \tag{7.16}
\]

which can be used to identify a system when the input and response are available. This concept can be considered in a sense of transfer function concept in Laplace domain, and will be a key element to develop the GLTM algorithm.

### 7.2.2 GLTM Method

GLTM is a method suggested by Shin et. al. [7] proposed as an alternative to CLTM. They showed that, in both of these methods, independently measured time-domain signals \( x(t) \) and \( y(t) \) are produced as a result of similar impacts on an aluminum sheet. Corresponding frequency spectra \( X(f) \) and \( Y(f) \) are produced by using these time-domain signals \( x(t) \) and \( y(t) \) and a fictitious linear time-invariant system is defined as in Fig.7.8, where \( X(f) \) and \( Y(f) \) are defined as in Eq.(7.17) and Eq.(7.18);
CHAPTER 7. GROUP DELAY-BASED LOCATION TEMPLATE MATCHING

Figure 7.8: Block diagram of fictitious LTI system. (Source [7])

\[
X(f) = \int_{-\infty}^{\infty} x(t)e^{j2\pi ft} dt
\]  
(7.17)

\[
Y(f) = \int_{-\infty}^{\infty} y(t)e^{j2\pi ft} dt
\]  
(7.18)

Finally, fictitious frequency response function \( H(f) \) is given by Eq.(7.19)

\[
H(f) = \frac{Y(f)}{X(f)} = \frac{Y(f)}{X(f)} e^{j(\phi_y(f) - \phi_x(f))} = |H(f)| e^{j\text{arg}H(f)}
\]  
(7.19)

Eq.(7.19) is used to measure the similarity between these two frequency spectra for each frequency component [7], \( X(f) \) and \( Y(f) \), where \( \text{arg}H(f) \) represents the shape relationship and \( |H(f)| \) represents the magnitude relationship. In other words, inspection of the phase structure and group delay of the frequency response function \( H(f) \) identifies the shape similarity between the two signals in time-domain, \( x(t) \) and \( y(t) \). Group delay of the frequency response function \( H(f) \) is given by Shin et. al.[7] as in Eq.(7.20):

\[
\tau_{xy}(f) = -\frac{1}{2\pi} \frac{\text{darg}H(f)}{df}
\]  
(7.20)

Note that group delay given in Eq.(7.20) is equal to zero for all frequencies when two compared time-domain signals \( x(t) \) and \( y(t) \) have the same shape without any time delay and match 100%. But if a pure time delay exists between these two time-domain signals which have the same shape, group delay remains constant with the same value for all frequency spectra. The last possible case is when two signals have difference in shapes, then group delay can not remain constant and varies across the frequencies [40]. Main criteria for GLTM algorithm is the deviation in group delay and it is used to identify degree of similarity. In the case that we have two identical signals, the group delay value approaches to constant values, and zero.

GLTM algorithm calculates the group delays first and then seeks for the least variance of group delay across the frequency axis pair. Variance of group delay (which is calculated with the help of Eq.(7.20)) is calculated with the following formula:
CHAPTER 7. GROUP DELAY-BASED LOCATION TEMPLATE MATCHING

\[ \text{Var}(\tau_{xy}(f)) = \sigma_{xy}^2(f) = E[(\tau_{xy}(f) - \mu_{xy}(f))^2] \]  

(7.21)

where \( \sigma_{xy}(f) \) and \( \mu_{xy}(f) \) represents standard deviation and mean value of group delay, \( \tau_{xy}(f) \).

7.3 Algorithm Implementation

Steps to develop GLTM algorithm is summarized as follows:

1. Define a grid to implement GLTM algorithm
2. Collect signals from different points in the grid
3. Build a location template using the frequency domain signals collected from the grid
4. Acquire the frequency domain signal from any point on the grid
5. Calculate the frequency response functions using the newly acquired signal and the ones already in the template
6. Calculate group delays for each Frequency Response Filter (FRF)

![Example group-delay vs. frequency. from Kistler accelerometer](image)

Figure 7.9: Example group-delay vs. frequency. from Kistler accelerometer
CHAPTER 7. GROUP DELAY-BASED LOCATION TEMPLATE MATCHING

Figure 7.10: Example variance and standard deviation of group-delays.

7. Calculate the variance of group delays for each FRF

8. Rearrange the grid with the calculated variances of group delay values

9. Compare the variance values. For example, it is easy to observe that the 3rd signal in Figure 7.9 has the least variance of group delay.

10. Determine the signal pair that has the least variance value

11. Declare the signal pair with the least variance as the source of the acoustic emission. In the case that is illustrated in Figure 7.9 and Figure 7.10, the point \( p_3 \) is the output of GLTM algorithm due to the least variance values.

7.4 Results and Comments

Similar to CLTM experiment, a 3x3 grid is used with 3 cm horizontal and 3 cm vertical distances between the nodes. 6 sets of data were collected from each node and the same data collection procedure was followed for both the single and 10 tap configurations. This table was built with the collection of 108 data sets collected in total in compliance with the other algorithms.

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Excitation</th>
<th>Accuracy</th>
<th>Distance</th>
<th>Grid Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kistler</td>
<td>Single Tap</td>
<td>0.907</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
<tr>
<td>e-Puck</td>
<td>Single Tap</td>
<td>0.851</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
<tr>
<td>Kistler</td>
<td>10 Taps</td>
<td>0.926</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
<tr>
<td>e-Puck</td>
<td>10 Taps</td>
<td>0.888</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
</tbody>
</table>

Table 7.1: Results of GLTM

An important advantage of GLTM over CLTM is that, GLTM uses frequency domain signals instead of time domain signals. That feature enables the use of group delay property, which
is less affected by the possible disturbances and noise in both reference and template signals. This property helps GLTM algorithm to deliver better results in terms of accuracy. With the help of GLTM algorithm, the majority of estimated acoustic emission sources are found correctly.
Chapter 8

Model Based Location Template Matching

8.1 Infinite Impulse Response Filter-based LTM

Infinite Impulse Response Filter (IIRF)-based LTM is another source localization technique which uses pattern recognition approach utilizing IIRF as a parameter for identification.

8.1.1 Overview

Popular acoustic emission methods which are aimed for sound recognition usually model how sensors (could be ears, any other microphone or even a speaker) respond to sound waves, or derivation of their application to other mechanical systems. In contrast to these methods, IIRF-based LTM uses dominant frequencies to model the actual system. It utilizes an IIR filter to model the system and compares the coefficients of the calculated IIRF. Comparison of the actual output and estimated output results in error in filter coefficients calculated through system identification tools. Comparison and minimization of this error leads to the source localization algorithm that is called IIRF-based LTM.

8.1.2 Details of Method

Unlike GLTM and CLTM, IIRF-based algorithm relies on IIR filter coefficients that expresses the actual structure. According to that model, an infinite impulse response filter that represents the system is calculated. As a result of comparison of the filter coefficients, location of
acoustic emission is estimated. Here, we would like to emphasize several concepts that will be helpful for the understanding of IIRF-based algorithm.

8.1.2.1 Finite and Infinite Impulse Response Filters

When a filter has a finite memory, this particular filter is called Finite Impulse Response (FIR) filter and similarly for the filters with infinite memory the term Infinite Impulse Response (IIR) is used. In general, FIR filters are non-recursive and IIR filters are recursive [41]. These properties of being ‘non-recursive’ or ‘recursive’ do not refer to length of memory but are used to describe realization of these filters.

At this point, we would like to give more details about IIR filters, since the algorithm is based on IIR filter. It should be noted that, IIRF is recursive and these filters generate phase distortion. This inevitable consequence is directly related to their structure [27]. We would like to mention two popular methods here to design an IIR filter: ‘Impulse-invariant’ method and ‘bilinear mapping’. While bilinear mapping method can avoid aliasing, impulse-invariant methods may be sensitive to aliasing [27].

Impulse-invariant method creates a filter whose impulse response sequence matches IIR of the corresponding analog filter. Although Impulse-invariant method is easy to design, it has inherent difficulties when designing high-pass and band-pass filters [27]. On the other hand, bilinear mapping method introduces frequency distortion (especially in high frequencies) but can be eliminated by ‘pre-wrapping’ [27].

As mentioned above, IIRF-based LTM relies on the IIR filter that represents mathematical model of the medium. First, an IIR filter is developed to represent the actual system. It is shown that profile of the output signals is unique both in time and in frequency domain [5] and this property is used to develop a source localization algorithm based on IIRF. Poletkin et. al. has successfully localized finger taps by the model they derived for LTM. With this technique, Poletkin et al. has also successfully converted hard surfaces into touch pads through the use of surface mounted sensors [5].

8.1.2.2 Generating Mathematical Model of the Structure

Poletkin et. al. has shown the mathematical model of surface vibration on solids for a rectangular plate in [5] using classical plate theory as the following:
Figure 8.1: Block diagram of IIRF-based LTM (Source [5])

and pressure at point with coordinates \((x, y)\) at time being \(t\) and vertical displacement \(w(x, y, t)\), at any point of the plate is given by the following formula:

\[
P(x, y, t) = D \nabla^4 w(x, y, t) + \mu \frac{dw(x, y, t)}{dt} + \rho L_z \frac{d^2w(x, y, t)}{dt^2}
\] (8.1)

where, \(\mu\) represents absorption coefficient, \(\rho\) represents density, \(L_z\) represents thickness of the plate, \(D = \frac{EL^3}{(1-\nu^2)}\), \(E\) represents Young’s modulus, \(\nu\) represents Poisson’s ratio and \(\nabla = \left(\frac{\partial^4}{\partial x^4} + 2\frac{\partial^4}{\partial y^4} + \frac{\partial^4}{\partial z^4}\right)\).

and in case impact location is given as \((x_0, y_0)\):

\[
P(x, y, t) = N_0(t)\delta(x - x_0)\delta(y - y_0)
\] (8.2)

Using these equations and proper Laplace transformation, transfer function of the physical system is given in [5] as the following:

\[
W_p(x, y, x_0, y_0, s) = \sum_{l=1}^{\infty} \sum_{b=1}^{\infty} \frac{4}{\rho L_x L_z L_y} \frac{W_{lb}}{s^2 + \bar{\mu} s + w_{lb}^2}
\] (8.3)

where

\[
W_{lb} = \sin(\alpha_l x)\sin(\beta_b y)\sin(\alpha_l x_0)\sin(\beta_b y_0)
\] (8.4)

\[
w_{lb} = \pi^2 \left(\frac{l^2}{L_x^2} \frac{b^2}{L_y^2}\right)^\frac{1}{2} \sqrt{\frac{D}{\rho L_z}}
\] (8.5)

\[
\alpha_l = \frac{\pi l}{L_x}
\] (8.6)
\[ \beta_b = \frac{\pi b}{L_y} \]  

(8.7)

and reduced coefficient of absorption is

\[ \tilde{\mu} = \frac{\mu}{\rho L_z} \]  

(8.8)

where \( l \) and \( b \) represent modes of wave propagation.

### 8.1.2.3 Generating Mathematical Model of Shock Accelerometer

Mathematical model for the Kistler Accelerometer is given by:

\[
M \frac{d^2 u(t)}{dt^2} + \zeta \frac{du(t)}{dt} + ku(t) = g(t)
\]

(8.9)

with accelerometer output \( u(t) \), input acceleration \( g(t) \), mass of accelerometer \( M \), damping coefficient \( \zeta \) and spring stiffness \( k \). This, corresponds to the transfer function that is given as the following:

\[
W_s(s) = \frac{Y(s)}{G(s)} = \frac{K_s}{T^2 s^2 + 2\zeta Ts + 1}
\]

(8.10)

where \( K_s \) is the gain, \( T = \sqrt{\frac{M}{k}} \) is the time constant and \( \zeta = \frac{\mu}{2\sqrt{MK}} \) is the relative damping coefficient.

### 8.1.2.4 Transfer Function of the System

With the detailed description above of both plate and accelerometer, the transfer function of the input-output system as per [5] becomes:

\[
TF = \frac{U_{out}(x_s, y_s, s)}{N_0(s)} = W_s(s)W_p(x, y, x_0, y_0, s)
\]

(8.11)

In [33], [42] and [5] it is reported that \( \pm 1.5 \) cm accuracy has been achieved using this method. A pattern recognition approach called IIRF-based LTM was also proposed in [5], which uses IIR-filter. In that study, Poletkin et. al. have utilized all-pole model on the received signal and determined dominant frequencies. This process has produced the minimized prediction error \( e(n) \), which represents the difference between predicted signal \( \hat{u}(n) \) and received \( u(n) \).

\[
e(n) = u(n) - \hat{u}(n)
\]

(8.12)
where
\[
\hat{u}(n) = \sum_{i=1}^{I} a(i) u(n - i)
\] (8.13)
and \( a(i) \) is the \( i^{th} \) filter coefficient, \( I \) is the prediction order.

The approach from the cited study can be outlined as follows;
- Calculate \( a(i) \), \( a_1(i) \) and \( a_2(i) \) filter coefficients, for two signals \( u_1(n) \) and \( u_2(n) \) from taps at two different points.
- Frequency-domain spectra are obtained using the Eq. (8.14) for the time domain signals \( u_1(n) \) and \( u_2(n) \).

\[
|U(e^{jw})|^2 = \sigma_e^2 \left| e^{jw} - e^{jw_1} \right|^2 \left| e^{jw} - e^{jw_I} \right|^2
\] (8.14)
where \( f_s \) is the sampling frequency and \( w = 2\pi f / f_s \) are dominant frequencies, \( \sigma_e^2 \) is the variance of \( e(n) \). \( U(e^{jw}) \) is the largest at the matched frequencies. Next, difference of the dominant frequencies are used to distinguish different sourced signals. In this goal, sum of the errors in filter coefficients is calculated through the formula;

\[
S_m = \sum_{i=1}^{I} (a(i) - a_{\text{template}}(i))^2
\] (8.15)

For each point that we have previously recorded, we computed the sum of errors, \( S_m \), between the template and reference and compare. Minimum \( S_m \) indicates the output of IIRF-based LTM.

### 8.1.3 Algorithm Implementation

Steps to develop IIRF-based LTM algorithm is summarized as listed below;

1. Define a grid to implement IIRF-based LTM
2. Collect signals from different points in the grid
3. Build a location template using the time domain signals collected from the grid
4. Acquire the time domain signal from any point on the grid
5. Define the order of IIRF
CHAPTER 8. MODEL BASED LOCATION TEMPLATE MATCHING

6. Calculate the filter coefficients \( a(i) \) and prediction error \( e(n) \) for each signal using the Levinson-Durbin algorithm.
   e.g., \([a_{38}, e_{38}] = \text{levinson}(x_{38}, 3);\)

7. Calculate summation of squares of the differences between filter coefficients from reference signal and template signal according to Eq. (8.15).
   e.g., \( S_{11} = (a_{ref}(1) - a_{11}(1))^2 + (a_{ref}(2) - a_{11}(2))^2 + (a_{ref}(3) - a_{11}(3))^2 + (a_{ref}(4) - a_{11}(4))^2;\)

8. Rearrange the grid with the calculated \( S_m \) values

9. Compare \( S_m \) values

10. Determine the signal pair which delivers the least \( S_m \)

11. Declare the signal pair with the minimum \( S_m \) as the source of the acoustic emission

8.1.4 Results and Comments

IIRF-based LTM method relies on an infinite impulse response filter which is used to model the mechanical system. Similar to CLTM and GLTM approaches, time domain signals are used to build the location template. With each signal in the template, we have approximated the physical system with an IIRF. We have calculated the filter coefficients depending on the order of IIRF. This yield to 54 sets of coefficients from 54 IIR filters. We have utilized these coefficients accordingly for localization purposes. Results are as shown below;

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Excitation</th>
<th>Accuracy</th>
<th>Distance</th>
<th>Grid Size</th>
<th>IIRF order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kistler</td>
<td>Single Tap</td>
<td>0.870</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>2</td>
</tr>
<tr>
<td>e-Puck</td>
<td>Single Tap</td>
<td>0.814</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>2</td>
</tr>
<tr>
<td>Kistler</td>
<td>10 Taps</td>
<td>0.888</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>2</td>
</tr>
<tr>
<td>e-Puck</td>
<td>10 Taps</td>
<td>0.851</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 8.1: Results of IIRF-based LTM with 2nd order IIRF
CHAPTER 8. MODEL BASED LOCATION TEMPLATE MATCHING

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Excitation</th>
<th>Accuracy</th>
<th>Distance</th>
<th>Grid Size</th>
<th>IIRF order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kistler</td>
<td>Single Tap</td>
<td>0.888</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>3</td>
</tr>
<tr>
<td>e-Puck</td>
<td>Single Tap</td>
<td>0.814</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>3</td>
</tr>
<tr>
<td>Kistler</td>
<td>10 Taps</td>
<td>0.888</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>3</td>
</tr>
<tr>
<td>e-Puck</td>
<td>10 Taps</td>
<td>0.851</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 8.2: Results of IIRF-based LTM with 3rd order IIRF

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Excitation</th>
<th>Accuracy</th>
<th>Distance</th>
<th>Grid Size</th>
<th>IIRF order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kistler</td>
<td>Single Tap</td>
<td>0.833</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>4</td>
</tr>
<tr>
<td>e-Puck</td>
<td>Single Tap</td>
<td>0.796</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>4</td>
</tr>
<tr>
<td>Kistler</td>
<td>10 Taps</td>
<td>0.888</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>4</td>
</tr>
<tr>
<td>e-Puck</td>
<td>10 Taps</td>
<td>0.796</td>
<td>3 cm</td>
<td>3 x 3</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 8.3: Results of IIRF-based LTM with 4th order IIRF

IIRF-based LTM utilizes the spectral peaks. This spectral-template matching technique reveals the uniqueness in each signal and makes it possible to be used for localization purposes. Even though the IIR filter coefficients which are slightly different for each signal pair, it is yet capable of highlighting the unique features within the signals. IIRF-based LTM takes advantage of this feature and delivers better accuracy than CLTM with a computationally less expensive process. It is worth noting that, even though it is less expensive to conduct, with the help of MATLAB, IIRF-based LTM is easy and fast to implement yet it is almost as accurate as the GLTM algorithm.

8.2 System Identification Based Location Template Matching

8.2.1 Overview

In general, system identification is used to generate a mathematical representation and understanding of a system as often done in the fields of system dynamics, control design and vibrations. This approach is used when inputs and outputs of a system are known but the mathematical model of the system is not available due to the complexity of the physical system or for the systems that require too much effort to be identified.

With this knowledge, here we generate transfer functions and used them to identify the
output of e-Puck in response to the taps generated by the stepper motor as in the previous experiments. Once we had these results, we also compared with the scenarios where we had a better accelerometer attached on e-Puck with wired connection.

8.2.2 Troubles Faced with Other Approaches when using e-Puck

In the literature, all three of the methods given above has been excessively studied, in a highly reliably constructed environment with strictly fixed sensor, well-isolated media and impacts which are generated using high precision tools such as impact hammers and high resolution servos with force and torque control. In this study, we aimed to test those methods on e-Puck robots in a more realistic environment with passively isolated medium and a relatively less accurate impact source. We have also changed the approach from fixed sensor and moving impact to moving sensor and a fixed impact. With these changes in search of a future mobile platform for acoustic emission localization, we could not reach the full extent of the capabilities of the methods but we have approximated the results from literature with certain error margin, slightly less accuracy and relatively better accuracy using Kistler’s accelerometer. When we changed the sensor from Kistler and wired DAQ to e-Puck and Bluetooth connection DAQ, we have noticed a dramatic change in terms of performance of each algorithm. With these in mind, we have reached a point where performance limitations of e-Puck robot has restricted the overall success.

8.2.3 Our Approach to Deliver Localization using e-Puck with modified IIRF-based LTM

During the experiments of the three mentioned algorithms, it was possible to use the algorithms to localize emission source with a degree of error due to the above summarized chain of changes but they could still be used for the purpose of localization. Moreover, accuracy of these methods can be further improved using the three e-Pucks at the same time.

Next, we wondered what if we had changed the IIRF-based LTM into something different with an additional transfer function estimation of the system and treat the transfer function as a representative of the actual system instead of IIRF. This change results in the change of comparison tool from the IIR filter to output of the estimated transfer function. Due to the poor DAQ capabilities of e-Puck, finding the transfer functions was not our priority but to let MATLAB approximate in terms of outputs. With the idea of using mathematical model of the system to guess system’s response from IIRF-based LTM, we have estimated the transfer functions for each signal pair, using
CHAPTER 8. MODEL BASED LOCATION TEMPLATE MATCHING

the previously logged outputs which has step function as inputs, to have the outputs from estimated transfer functions. Later, output from the transfer function is fed into the GLTM algorithm and the output is calculated. In some sense the transfer function is used to clear the output signal before it is compared with a test signal, with the help of GLTM algorithm.

8.2.4 Algorithm Implementation

Steps to develop System Identification-based LTM algorithm can be summarized as listed below;

1. Define a grid to implement System Identification-based LTM
2. Collect signals from different points in the grid
3. Build a location template using the time domain signals collected from the grid
4. Acquire the time domain signal from any point on the grid
5. Use MATLAB’s System Identification Toolbox IDE to estimate mathematical model of the system for each node using step functions as inputs.

![Figure 8.2: Simulink diagram for step input generation](image)

6. Calculate the outputs according to the estimated models
7. Treat the outputs from the model as measurements from the grid in GLTM algorithm to determine the least group delay of the each signal
8. Determine the least in group delay and declare that as the output of the algorithm.
CHAPTER 8. MODEL BASED LOCATION TEMPLATE MATCHING

8.2.5 Results

Due to the complexity of e-Puck’s mechanical configuration we have chosen another approach as explained above. Once we gathered the transfer functions, we fed the output of transfer function estimation as the output of the system at the selected nodes in the spirit of LTM just like in the IIRF-based LTM algorithm. We have tried this modified IIRF-based LTM/GLTM approach using Kistler accelerometer on top of the e-Puck robot. Results of this approach was close in terms of performance for both Kistler and e-Puck instrumentations. Of course, results have changed into a better situation with the Kistler yet only marginally. We believe this occurs due to the complex structure of e-Puck robot which may not be perfectly represented by a transfer function.

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Excitation</th>
<th>Accuracy</th>
<th>Distance</th>
<th>Grid Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kistler</td>
<td>Single Tap</td>
<td>0.888</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
<tr>
<td>e-Puck</td>
<td>Single Tap</td>
<td>0.833</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
<tr>
<td>Kistler</td>
<td>10 Taps</td>
<td>0.888</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
<tr>
<td>e-Puck</td>
<td>10 Taps</td>
<td>0.796</td>
<td>3 cm</td>
<td>3 x 3</td>
</tr>
</tbody>
</table>

Table 8.4: Results of Transfer Function-based LTM

We believe the performance of this proposed method is affected by the inaccuracy of stepper motor in terms of production of taps. Since every time we tap onto the acrylic plane, the input and the noise is changing we ended up with different signals both acquired and estimated. These differences yield to the performance of the proposed algorithm. Also, it should be noted that usage of advanced filters can improve signal quality. We believe that improvement on these factors could help to deliver better performances. Finally, it also should be noted that a drawback of the proposed, system identification based approach is that, it is computationally more expensive than the other three methods.
Chapter 9

Conclusions and Future Work

9.1 Conclusions

We have investigated four different LTM algorithms to measure the acoustic emission localization capabilities of the e-Puck robot. We have seen that GLTM and IIRF-based LTM algorithms delivered better results while CLTM was barely usable with e-Puck robots. We have managed to approximate the results in literature on the idea behind IIRF-based LTM, we have proposed an additional method that is in fact a modified version of IIRF approach in connection with GLTM and requires the extraction of transfer functions calculated with the help MATLAB’s System Identification Toolbox. We have seen that even though it is computationally more expensive than other methods, this approach produces close results in terms of accuracy when we compared to the other methods.

With this work, it is shown that mobile robots could be used as an alternative to fixed sensors. Mobile platforms which have accelerometers on-board could be used in a sense of sensor fusion for better accuracy. With the utilization of mobile platforms, it is possible to track moving emission sources or these platform could be used to enhance resolution by moving closer to objects causing acoustic emission. These systems hence have advantages in terms of flexibility in both operation and application.

9.2 Future Work

With the improvements achieved by this work, potential is shown to the extent time permitted. It should be noted that even though some properties of the signals are used for localization
CHAPTER 9. CONCLUSIONS AND FUTURE WORK

intentions within the given margins, there are some other properties that have not been covered and have not been paid attention. For those properties within the signals, supervised and unsupervised machine learning algorithms such as, Perception Learning Algorithm, Pocket Algorithm and Linear Regression can be used.

We think that a proper implementation of machine learning algorithms, will allow other statistical properties to be used. This additional feature of the acoustic emission signal should be helpful to achieve more advanced and smart applications with better accuracy. If these tools are accompanied by other advanced signal processing capabilities, these algorithms can be useful to get better results for acoustic emission studies conducted using e-Puck robots. It also should be noted that, utilization of multiple e-Puck robots can help to gather improved results in a sense of sensor fusion.
Bibliography


BIBLIOGRAPHY


[14] “Stepper motor website.” [http://www.adafruit.com/products/324?gclid=CjwKEAjw9PioBRDdpqy0-ofG3DgSJAACe5NE50Rdrootwz4ZwczUPceE/-SRYa8w7iR4-DRe8fPdhQWRoC3b3w_wcB](http://www.adafruit.com/products/324?gclid=CjwKEAjw9PioBRDdpqy0-ofG3DgSJAACe5NE50Rdrootwz4ZwczUPceE/-SRYa8w7iR4-DRe8fPdhQWRoC3b3w_wcB) Last Accessed: 2015-06-23.


