Ultrasonic Sensor Platforms for Non-Destructive Evaluation

By

Gordon Dobie

In fulfilment of the requirement
for the degree of Doctor of Philosophy.

Centre for Ultrasonic Engineering
Department of Electrical and Electronic Engineering
University of Strathclyde

Copyright

Acknowledgement

I would like to thank Professor Gordon Hayward of the Centre for Ultrasonics Engineering at the University of Strathclyde for the opportunity to conduct this research. Many thanks to Gareth Pierce, Walter Galbraith, Liam Mackenzie and Rahul Summan for their technical input throughout this project. Furthermore, I am most grateful for the support of the Research Centre for Non-Destructive Evaluation (RCNDE), in particular the Development Manager, Peter Thayer.

I would like to thank my wife Gillian and parents William and Susan for their support throughout this project and our daughter Emily, whose arrival in October 2009 provided good motivation to finish this thesis.

Abstract

Robotic vehicles are receiving increasing attention for use in Non-Destructive Evaluation (NDE), due to their attractiveness in terms of cost, safety and their accessibility to areas where manual inspection is not practical. A reconfigurable Lamb wave scanner, using autonomous robotic platforms is presented. The scanner is built from a fleet of wireless miniature robotic vehicles, each with a non-contact ultrasonic payload capable of generating the A_0 Lamb wave mode in plate specimens. An embedded Kalman filter gives the robots a positional accuracy of 10mm.

A computer simulator, to facilitate the design and assessment of the reconfigurable scanner, is also presented. Transducer behaviour has been simulated using a Linear Systems approximation (LS), with wave propagation in the structure modelled using the Local Interaction Simulation Approach (LISA). Integration of the LS and LISA approaches were validated for use in Lamb wave scanning by comparison with both analytical techniques and more computationally intensive commercial finite element/difference codes. Starting with fundamental dispersion data, the work goes on to describe the simulation of wave propagation and the subsequent interaction with artificial defects and plate boundaries.

The computer simulator was used to evaluate several imaging techniques, including local inspection of the area under the robot and an extended method that emits an ultrasonic wave and listens for echos (B-Scan). These algorithms were implemented in the robotic platform and experimental results are presented. The Synthetic Aperture Focusing Technique (SAFT) was evaluated as a means of improving the fidelity of B-Scan data. It was found that a SAFT is only effective for transducers with reasonably wide beam divergence, necessitating small transducers with a width of approximately 5mm. Finally, an algorithm for robot localisation relative to plate sections was proposed and experimentally validated.

Table of Contents

List of Figures

List of Acronyms

General

Acronym	Definition
CUE	Centre for Ultrasonic Engineering. The authors research group.
DMEM	Department of Manufacture and Engineering Management. An engineering department at the University of Strathclyde.
NDE	Non-Destructive Evaluation.
RSA	Remote Sensing Agent. The miniature robotic vehicles described in this work.

Electrical Engineering

Acronym	Definition
IC	Integrated Circuit. A circuit packaged as a single chip.
LED	Light Emitting Diode.
FET	Field Effect Transistor.
MOSFET	Metal Oxide Field Effect Transistor.
PID	Proportional Integral Differential controller. A standard approach to closed loop control. PID is discussed in Chapter 2
PWM	Pulse Width Modulation.
RF	Radio Frequency.

Robotics

Acronym	Definition
EKF	Extended Kalman Filter. An algorithm used in robotic positioning. The EKF is discussed in detail in Chapter 3

GPS
Global Positioning System. In this work GPS is used to represent any global positioning system and not necessarily the satellite GPS system created by the United States.

SLAM
Simultaneous Positioning and Mapping Algorithm. An algorithm that uses sensor data to simultaneously generate a map of the robots environment and then position the robot relative to the map.

Software Engineering

Acronym
Definition

ADC
Analogue to Digital Convertor.

C
A general purpose computer programming language, typically used to program microcontrollers.

C++
An object oriented extension of C, typically used as a medium-level language, providing relatively high performance.

C#
C# was created to work in the Common Language Infrastructure, which in converted to an intermediate language and run in the Common Language Runtime CLR. This execution environment isolates the program from the native hardware, improving memory and thread management and garbage collection. This provides a convenient development platform and robust software at the cost of reduced performance.

CPU
Central Processing Unit.

DMA
Direct Memory Access. Provides direct access to memory independently of the CPU.

Flash Memory
Non-violate computer memory used store data, such as programs, even when the system is powered down.

GUI
Graphical User Interface.

I2C
Inter-Integrated Circuit. A two wire master-slave communications bus, generally used to communicated between ICs on a circuit board. A master may be connected to several slaves on a single bus. The master selects individual slaves using an address byte at the start of each communication.

IO	Input / Output. Many microcontroller ports can be configured as input or output in software.
IP	Internet Protocol. The communications network used for networks and the internet.
Linux	An open source Unix-like computer operating systems based on the Linux kernel.
RAM	Random Access Memory. Violate computer memory used to store data at run time.
SSH	Secure SHell. A secure network communications protocol.
SPI	Serial Peripheral Interface. Another communications protocol. This is generally fast, but is relatively low level, dealing in individual bytes, rather than packets of data.
UART	A Universal Asynchronous Receiver/Transmitter. Electronic hardware that handles standard serial communication. Common for communication between embedded systems and the RS-232 (serial) port on a PC.
USB	Universal Serial Bus. A high speed serial interface available on modern computers.
Wi-Fi	A wireless network protocol, formally known as 802.11b/g/n.

Ultrasonics

Acronym	**Definition**
A_0	Zero-order antisymmetric Lamb wave mode.
CY1301/HY1300	Araldite Epoxy, manufactured by Huntsman [2].
EMAT	Electrical Mechanical Acoustic Transducer. An ultrasonic transducer that is typically created from a coil and permanent magnet.
FE	Finite Element. A modelling technique.
FD	Finite Difference. A modelling technique.

FOV	Field of View.
FTP	Frequency Thickness Product.
FMC	Full Matrix Capture.
LISA	Local Interaction Simulation Approach. A technique for modeling wave propagation. LISA is discussed in Chapter 5.
LSM	Linear Systems Model. A 1D model of piezoelectric transducers. The LSM is discussed in Chapter 5.
PZT-5A	Piezoelectric material, supplied by Morgan Electroceramics [3].
PZT-5H	Piezoelectric material, supplied by Morgan Electroceramics [3].
S_0	Zero-order symmetrical Lamb wave mode.
SAFT	Synthetic Aperture Focusing Technique. Algorithm to improve the resolution of ultrasonic scans, discussed in Chapter 7
SIRT	Simultaneous Iterative Reconstruction Technique. A algebraic reconstruction technique used in tomography.
SNR	Signal to Noise Ratio.
TFM	Total Focusing Method. An imaging algorithm.
TOF	Time of Flight.
UT	Ultrasonic Testing.

Chapter 1

Introduction

The industrialised world is full of structural assets that need to be periodically inspected to prevent unscheduled and costly outages or catastrophic failures. Such failures can lead to large scale environmental disasters and loss of life. These inspections are generally referred to as **N**on-**D**estructive **E**valuation (**NDE**), which determines the structural integrity of a component without testing it to breaking point. It is achieved by testing the component for degradation, using specific measurements that detect problems such as cracks, corrosion, or impact damage. NDE can be done in-situ, often while the component is in use.

NDE is an activity that underpins much of modern society, including energy production and delivery, public and private transportation (air, ground and sea), national infrastructure, and many sectors of commercial manufacture. Safety, economic and legislative factors are all driving a growth in NDE equipment which was valued at \$1.4 billion in 2003 and estimated to have grown to \$1.8 billion by 2008 [4]. There is a continuing drive to improve the accuracy, to improve safety, and to reduce the cost of NDE.

Inspections are generally performed manually, with trained operators scanning the structure with handheld probes. ‘Human factors’ are often the weakest component in NDE inspections [5] with overwork, lack of sufficient training and monotonous inspections leading to inconsistent quality [6]. Furthermore, human access to the inspection site is often difficult or intrinsically unsafe. Inspection sites are commonly located in inconvenient locations, partially obstructed, or in uncomfortable working environments. Access may require extensive scaffolding, or plant shutdowns. In the most extreme cases, like those found in the nuclear industry, radioactive or chemical contamination prohibits human access altogether. The UK Health and Safety Executive (HSE) is a full member of the UK’s Research Centre for Non-Destructive

Evaluation [7], a consortium of fourteen end users of NDE including Airbus, Alstom Power, BP, British Energy, National Nuclear Laboratory, Defence Science and Technology Laboratory (Dstl), E.ON Engineering, Network Rail, Petrobras, Rolls Royce, RWE npower, Serco, Shell and Tenaris. The full membership of the HSE highlights the importance of NDE for disaster prevention and the prevalence of safety concerns when preforming NDE.

Making the inspection site safe by erecting scaffolding or shutting down sections of a plant poses a significant financial burden. In the oil and gas industries access costs can typically be in excess of 70 per cent of the total budget, easily exceeding £100,000 per inspection. In addition to access, the inspection of large structures, such as storage tanks, is a labour intensive task leading to prohibitively high costs.

Miniature **R**emote **S**ensing **A**gents (**RSA**s) are well suited for the inspection of many structures and installations. Onboard power and wireless communications obviates the need for umbilical cabling maximising access and mobility. Their small size and novel traction makes them particularly useful for areas with limited access. It allows them to crawl over the surface of a structure, along pipes, and into small spaces. Moreover, the concept can be extended to a fleet of heterogenous robots which improves the functionality, reliability and flexibility of the system. The fleet consists of several RSAs, each carrying one of a plethora of NDE inspection measurement payloads based on ultrasonic, electromagnetic or visual modalities. Flexibility is achieved by adjusting the size and consistency of the fleet. For example, one robot may be used to inspect a pipe section, five robots may be used to inspect a small boiler and one hundred robots may be used to inspect the hull of a ship. Reliability is achieved through division of labour and redundancy; no individual robot is crucial to the inspection. The system can be optimised in real time, based on new inspection results or a change in operating conditions. The various inspection payloads can be used in a complementary manner. For example, visual inspection provides detailed information about the surface, whereas ultrasound can penetrate into the material, albeit with coarser resolution. Lastly, the individual robots can be made to be low-cost and even disposable, with the complex capabilities emerging through the use of a fleet, rather than through individual complexity.

1.1 Current State of the Technology at Strathclyde University

The section introduces the inspection system developed at Strathclyde University. The background review in Chapter 2 puts this work into context in the wider, international scene.

1.1.1 Miniature Robotic Vehicles

The RSAs have a volume of 14x14x10 cm and weigh 610g. Magnetic wheels facilitate traction to ferromagnetic structures allowing them to perform in-situ inspection of complex three dimensional structures. Each RSA is equipped with a embedded computer that provides extensive local processing and semi-autonomous decision making. A WiFi communication link provides high bandwidth communications for fast transfer of sensor data either between robots or back to a host computer. Each robot is equipped with several status monitoring sensors, including optical encoders and inertial sensors. The robots are powered by a high capacity battery which provides four hours of autonomous run-time. Figure 1.1 shows two RSAs performing an inspection of a 25m steel chimney.

1.1.2 Inspection Payloads

At the time of writing, the RSAs can carry one of four possible NDE payloads. These include:

1. A miniature camera capable of producing high resolution colour images, such as those shown in Figure 1.1. These can be combined to create visual maps of the structure under test. For more information see Chapter 2.

2. A differential eddy current probe that is particularly effective for investigating service induced fatigue and stress corrosion cracks in conducting materials [8].

3. A magnetic flux leakage payload that is effective in detecting material loss due to corrosion, pitting or gouging in ferromagnetic materials. This payload elegantly makes use of the magnetic field generated by the robot's magnetic wheels [8].

4. An non-contact ultrasonic inspection payload that uses guided waves to rapidly screen plate like structures. For more information see Chapter 4.

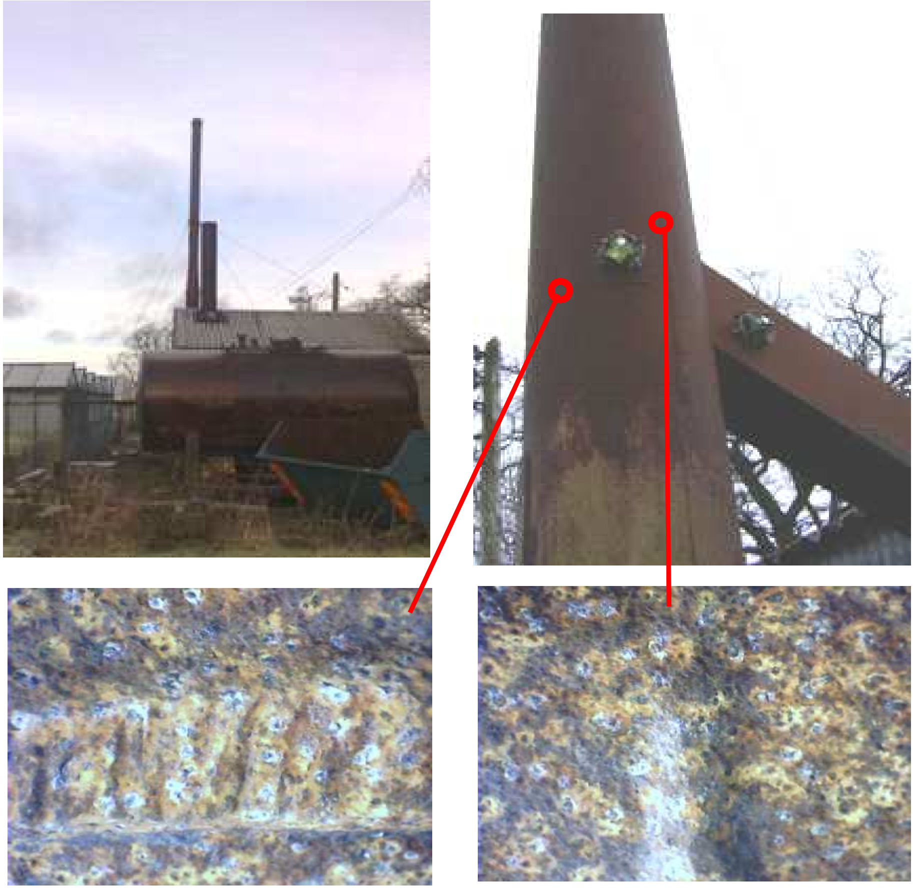


Figure 1.1: Two RSAs performing a visual inspection of a 25m steel chimney

1.1.3 Robot Positioning

Accurate robot positioning is critical for the realisation of an RSA inspection system. It is required for robot navigation, to identify the location of any defects, to facilitate reliable scan patterns and to combine aggregated sensor data into a unified set of results. Accurate positioning is one of the greatest challenges facing autonomous robotic systems and is forming a considerable portion of ongoing research. Significant progress has been made in combining global positioning measurements with local odometery using probabilistic approaches such as the Kalman Filter.

1.1.4 Collaborative Teams of Intelligent Robots

One of the most exciting applications of the technology is using teams of robots to optimise the inspection. In addition to improved fault tolerance and the facility for a division of labour, the robots can actually work together to improve the scanning procedure. A single robot may map the surface using ultrasonic ranging and then instruct other robots with complementary payloads to inspect certain regions, leading to an intelligent fusion of inspection methodologies. Ultrasonic imaging is discussed in Chapter 7. Friedrich [8] has demonstrated combining data sets from different sensor payloads using probabilistic and evidence based approaches (Bayesian and Dempster-Shafer). Ongoing research is working to use information gathered from fused data sets to dynamically update robot objectives.

1.2 Lamb Waves

Lamb waves are a form of ultrasonic guided wave that propagate through plate like structures [9]. They offer significant potential for the following reasons:

1. Lamb waves can be generated with non-contact air coupled angled transducers [10]. This obviates the need for a constant supply of coupling fluid which is critical for mobile, umbilical free scanning.

2. When generating the zero-order antisymmetric Lamb wave mode in the dispersive region, the excitation angle and operating frequency of the angled transducer are dependent on the specimen's material thickness. If the excitation frequency is constant, the thickness can be calculated by sweeping the incident angle. Alternatively the incident angle can be fixed and the frequency swept.

3. It is possible to rapidly scan large areas by propagating Lamb waves over long distances.

1.3 Aims of this Thesis

This thesis forms part of a long-term research commitment into robotic NDE by the **C**enter for **U**ltrasonic **E**ngineering (**CUE**) at the University of Strathclyde. Previous work [8] has introduced the concept of miniature robotic vehicles for NDE and has investigated combining multiple sensor payloads. Concurrent projects are investigating additional sensor payloads, such as eddy current arrays and visual payloads with automatic defect classification or automatic weld tracking, visual based positioning and ultrasonic communication through solid interfaces. The focus of this work was to evaluate the use of non-contact ultrasound for NDE inspection on mobile RSA platforms.

The creation of a fully autonomous fleet of reconfigurable RSAs necessitates careful analysis with regard to vehicle positional accuracy, the nature of the structure under test and the optimal combination of sensor units. Arguably, this is best achieved with the aid of a computer simulation which is capable of accurately replicating the entire system. This confers additional advantages for data interpretation and vehicle guidance. To be practically useful, full simulation in three dimensional (3D) space is required. This encompasses sensor behaviour (including interfacing), RSA positional information, structural form and defect modelling. Any simulator has to be capable of modelling transducer characteristics, wave propagation and interaction with meaningful synthetic artifacts, in addition to variations in the positional certainty of individual RSAs within the fleet. The simulator also has to be reasonably interactive in order to assist the user with algorithmic development and system design issues. To this end, the PhD was divided into two threads, simulation and hardware. The final goal was to implement techniques developed through simulation into the hardware platform.

The hardware thread consisted of creating a robust RSA platform which included miniature robotic vehicles, an ultrasonic payload, a robot positioning algorithm and supervisory software. These aims are shown graphically in Figure 1.2 and are discussed in more detail below.

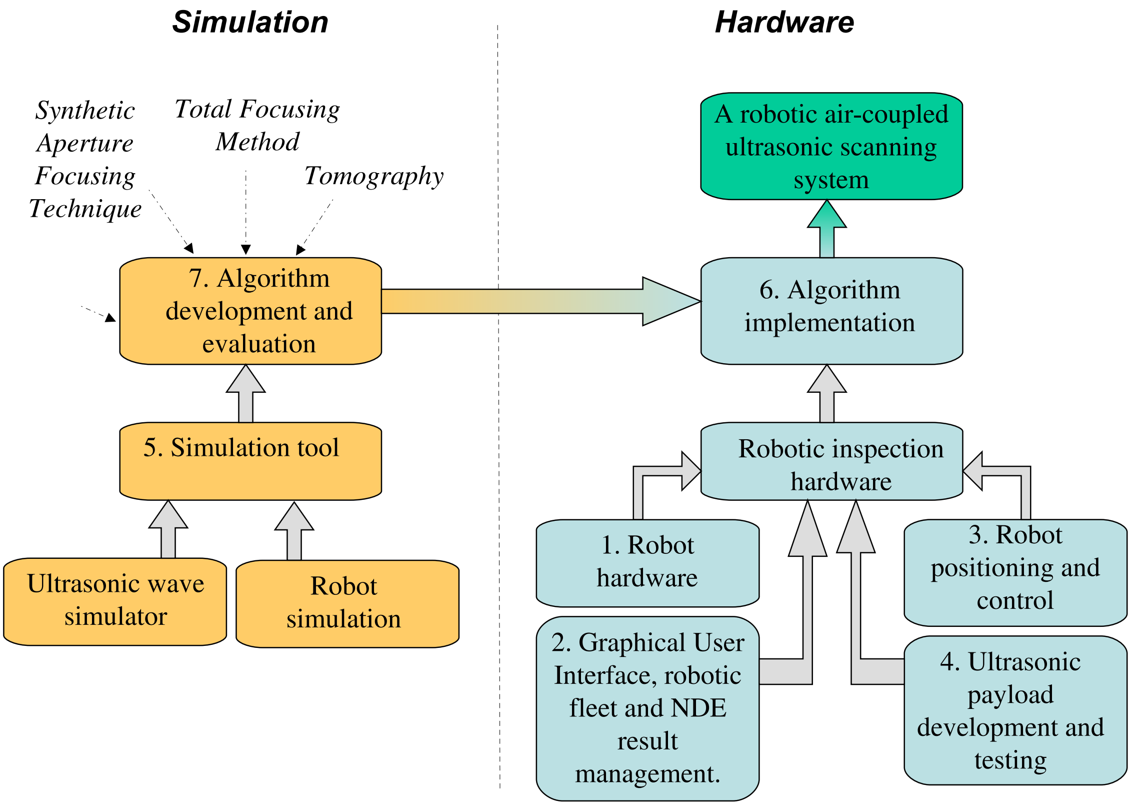

Simulation
Hardware
Synthetic Aperture Focusing Technique
Total Focusing Method
Tomography
7. Algorithm development and evaluation
5. Simulation tool
Ultrasonic wave simulator
Robot simulation
A robotic air-coupled ultrasonic scanning system
6. Algorithm implementation
Robotic inspection hardware
1. Robot hardware
2. Graphical User Interface, robotic fleet and NDE result management.
3. Robot positioning and control
4. Ultrasonic payload development and testing

Figure 1.2: Flow diagram showing the thesis objectives

1.3.1 Thesis Structure

With reference to Figure 1.2.

1. Chapter 2 discusses the design and realisation of a development robot for effective research, focussing on reliability, flexibility, modularity, high performance computational capabilities and high bandwidth communications.
2. Chapter 2 also discusses the design and implementation of a **G**raphical **U**ser **I**nterface (**GUI**) and supervisory software to manage a fleet of robots. The GUI also manages NDE measurements generated from multiple robots.
3. Chapter 3 discusses the implementation and evaluation of a Kalman Filter for robot positional tracking. This included measurement of benchmark results that can be used for comparison with alternative algorithms.
4. Chapter 4 discusses the design and evaluation of an embedded air-coupled ultrasonic sensor payload.
5. Chapter 5 discusses the creation and validation of a simulation tool that incorporates ultrasonic generation, propagation and reception, as well as inaccurate robot positioning.
6. Chapter 6 describes the use of the supervisory software and an ultrasonic equipped development robot, optimally positioned, to facilitate air-coupled scanning.
7. Chapter 7 describes the use of the simulation tool to assess the suitability of conventional ultrasonic imaging algorithms to the robotic system. A B-scan algorithm was experimentally implemented.

1.4 Contributions to Knowledge

A NDE scanner comprising a fleet of miniature robotic vehicles is a complex system with numerous interacting subsystems that include the robot hardware, the robot control systems, the robot's embedded logic and reasoning, the supervisory software, the Graphical User Interface, the robot positioning system (hardware and algorithms), the NDE payloads and the result management and interpretation software. Unless due care is taken to ensure a high level of robustness for each subsystem, particularly the core robotic hardware and supervisory software, the resulting work will be nothing more than a 'house of cards', that does not work reliably and cannot be further developed or expanded. For this reason, a considerable portion of this thesis is spent on the system design. A core contribution of this work is a flexible and robust robotic platform on top of which future research can be built.

A Kalman filter was integrated for robot positional tracking. The Kalman filter provides a probabilistic framework to optimally combine local positioning measurements such as those from odometry with global positioning measurements [11]. The resulting positional estimate is more accurate than any of the individual measurement sources. Kalman filters have been in use since their conception by Rudolf Kalman in 1960 [12] and have been applied to robotics in numerous occasions [11, 13, 14, 15]. The implementation of this filter is not in itself new. However, positional tracking is of critical importance to robotic NDE scanning and the rigorous implementation is not a trivial exercise. A series of experimental results provide a benchmark from which to compare more novel methods, something which is crucial for future research.

The creation of a non-contact ultrasonic payload that is physically small enough to fit on a miniature robotic vehicle without the need for external circuitry or power is one of the greatest technical achievements of this work. This provides a tangible new technology that could be exploited both academically in novel research and commercially as a new NDE measurement tool.

The ultrasonic payload was complemented by the implementation and validation of an efficient (and hence quick) 3D ultrasonic wave generation, propagation and reception simulation tool. This provided a useful tool for exploring the full potential of using miniature robotic vehicles with air-coupled ultrasound, as well as investigating ultrasonics in general.

'Local pitch catch' is the simplest implementation of the ultrasonic payload. It involves mounting transmit and receive transducers side by side on a single RSA, generating, then measuring a Lamb wave transmitted over the minimum possible distance. This tests the ability of the area under the transducers to propagate a specific type of Lamb wave. The transducers are tuned to nominal plate thickness and any change from this results in a loss of signal which can be interpreted as a defective region. The resolution is approximately equal to the combined surface area of the two transducers (40 × 20 mm for two square 20 mm wide transducers). Practically implementing this scanning methodology presented two issues. Firstly, there is uncertainty in the robot position and secondly measurement regions often overlap. An algorithm was devised that combined all measurements in an optimal manner.

The applicability of the state of the art ultrasonic imaging algorithms was evaluated with respect to the robotic fleet. It was found that tomographic techniques were ineffective for a scanner comprised of mobile sensors. Tomography requires a vast number of spatially distributed measurements, which translates into impractically long robot scan paths. In general, tomographic algorithms do not need to make any assumptions about the sample under test. However, by making some reasonable assumptions it is possible to drastically reduce the number of measurements required. By assuming that the sample is a uniform plate with a number of point reflectors, it is possible to image the plate using Lamb wave B-scans. To perform an ultrasonic B-scan, the transducers are moved through a series of points along one edge of the sample, at each point emitting a pulse, then listening for echos. B-scanning offers the potential for rapid scanning.

The **S**ynthetic **A**perture **F**ocusing **T**echnique (**SAFT**) [16] is an algorithm that can improve the resolving power and signal to noise ratio of B-scans by considering a series of measurements as a single synthetic aperture. A core contribution of this work is the detailed evaluation of SAFT as a means of post processing RSA generated B-scans. SAFT requires a wide field of view from each transducer, but it was found that an air-coupled ultrasonic transducer would need to be relatively small (in the order of 5mm wide) for the scan to benefit from SAFT processing. Reducing the size of the transducer, reduces the active area, reducing power input/output, reducing **S**ignal to **N**oise **R**atio (**SNR**) and hence image quality.

Additionally, SAFT was found to be sensitive to mobile transducer positioning inaccuracies. It was found that by artificially limiting the transducer field of view as defined in the SAFT algorithm, it was possible to reduce the sensitivity to positional inaccuracies at the cost of limiting resolving power. Guidelines are suggested for the applicability of SAFT depending on the field of view of the transducer and the positional accuracy of the robot platform.

A minimal level of context awareness is required to automatically B-scan a sample. Ideally, the robot should scan parallel to the edge of the plate section. A novel algorithm was developed to automatically position a robot on a rectangular plate of unknown dimensions. This was achieved using a radial arrangement of pulse-echo Lamb waves to identify plate edges. The algorithm was shown to be robust in plates containing defects, and hence additional reflectors.

The local pitch catch and pulse echo algorithms were both implemented experimentally with results presented in Chapters 6 and 7 respectively. The final contribution was an appraisal of the experimental system, how it can be improved and how the research can be taken forward.

1.5 Publications Arising from this Thesis

G. Dobie, W. Galbraith, M. Friedrich, S.G. Pierce and G. Hayward, “Robotic Based Reconfigurable Lamb Wave Scanner for Non-Destructive Evaluation”, *Proc. 2007 IEEE International Ultrasonics Symposium, New York*, 2007.

G. Dobie, A. Spencer, S.G. Pierce, W. Galbraith, K. Worden and G. Hayward. “Simulation and Implementation of Ultrasonic Remote Sensing Agents for Reconfigurable NDE Scanning”, *Proc. 2008 Review of Progress in Quantitative Nondestructive Evaluation, Chicago*, 2009.

R. Summan, G. Dobie, J. Hensman, G. Pierce and K. Worden, “A Probabilistic Approach to Robotic NDE Inspection”, *Proc. 2009 Review of Progress in Quantitative Nondestructive Evaluation, Chicago*, 2009.

G. Dobie, A. Spencer, K. Burnham, S. G. Pierce, K. Worden, W. Galbraith and G Hayward, “Simulation of Ultrasonic Lamb Wave Generation, Propagation and Detection for an Air Coupled Robotic Scanner”, Ultrasonics, 2009, *under review.*

M. Friedrich, G. Dobie, C. Chan, S. G Pierce, W. Galbraith, S. Marshall, and Gordon Hayward, “Miniature Mobile Sensor Platforms for Condition Monitoring of Structures”, IEEE Sensors Journal - Special Issue on Sensor Systems for Structural Health Monitoring, Vol. 9, Issue 11, pp. 1439-1448.

S. G. Pierce, G. Dobie, R. Summan, L. Mackenzie, J. Hensman, K. Worden, G. Hayward, ”Positioning Challenges in Reconfigurable Semi-autonomous Robotic NDE Inspection”, *Accepted for SPIE Smart Structures and Materials + Nondestructive Evaluation and Health Monitoring, San Diego, 2010*

Chapter 2

A Development Platform for Miniature Robotic NDE Research

This chapter discusses the design and realisation of the RSA development robot. It begins with a an overview of the field discussing both robotic NDE and miniature robotic vehicles. The remote sensing agent system is detailed, including mechanical design, electronics and firmware. Several peripherals are discussed, including an ultrasonic global positioning system, a visual positioning system and a visual NDE payload. The system is integrated through a PC software package that acts as a Graphical User Interface for fleet control and the presentation of NDE results. Subsequent chapters discuss robot positioning and the ultrasonic NDE payload.

The robotics research in CUE brings together the fields of swarm robotics and NDE. It is fundamentally different to existing robotics NDE in that it aims to use a fleet of miniature autonomous robots, rather than a single, large, tethered robot. This presents interesting challenges in terms of miniaturisation of sensor payloads, wireless operation and the positioning and control of the robotic fleet. Although there have isolated examples of the swarm robotics community investigating NDE [17, 18], NDE was only used to demonstrate swarm behaviour; the authors did not provide practical NDE solutions. A notable difference between this research and general swarm robotics research is that in swarm robotics the solutions do not use a centralised controller for the fleet, instead intelligent behaviour emerges from the swarm. We use a central control for coordinated inspection. In addition to this, we consider NDE as an intrinsic part of the system, allocating considerable resource to sensor design and the interpretation of measurements.

The specific contribution of this chapter is the creation of the RSA development system, specifically designed for the research needs of CUE. The system includes the vehicles hardware, the electronics, the firmware and the Graphical User Interface (GUI) to control the fleet.

2.1 Robotic Non-Destructive Evaluation

Over the last decade, the advantages of combining NDE inspections with robotic technology has become well established. Several groups have developed prototype robotic vehicle designs, the majority of which are designed for specific inspections including ships hulls [19, 20], storage tanks [21], aircraft [22, 23] and pipework, both internal [22, 24] and external [25].

Since NDE must be performed on 3D structures, NDE robots require the ability to climb over the surface of the test specimen. The design of a climbing robot is driven by the method of traction. Traction methods include magnetic [26, 8, 27], grippers [25], vacuum cups [28, 22, 21, 29, 30], electro-adhesion [31] and gecko inspired [32, 33]. Electro-adhesion and gecko inspired techniques are still immature and are not reliable enough for off-the-shelf use. The majority of NDE robots favour magnetic and vacuum cup technologies.

Permanent magnets offer a high adhesive force whilst requiring no power. They can be made very small and even integrated into the structure of the robot [8]. Magnetic traction is ideal for ferromagnetic structures that are abundant in the oil and gas industry, but is of limited use for inspecting the aluminium or carbon-fibre found in the aerospace industry or the austenitic stainless steel found in the nuclear industry. In these cases, vacuum cup based technologies are used, but these require a smooth surface and are a constant power drain. Most robots that use vacuum cup traction require an umbilical cable for power.

A commercial example of a NDE crawler using magnetic traction is the Scorpion series from Silverwing [26], a UK based company specialising in NDE equipment. The crawlers are approximately 385 × 222 × 102 mm and perform dry and water coupled ultrasonic inspection. An umbilical provides communications and power. The Scorpion DCP is shown in Figure 2.1.

The Jet Propulsion Laboratory in California has developed the Multifunctional Automated Crawling System (MACS) [28] that uses suction cups for non-magnetic surfaces. This MACS is also shown in Figure 2.1. It is specifically designed for aircraft inspection, such as external inspection of the C-5. The platform is approximately 300 × 100 mm and employs ultrasonic motors for mobility and suction cups

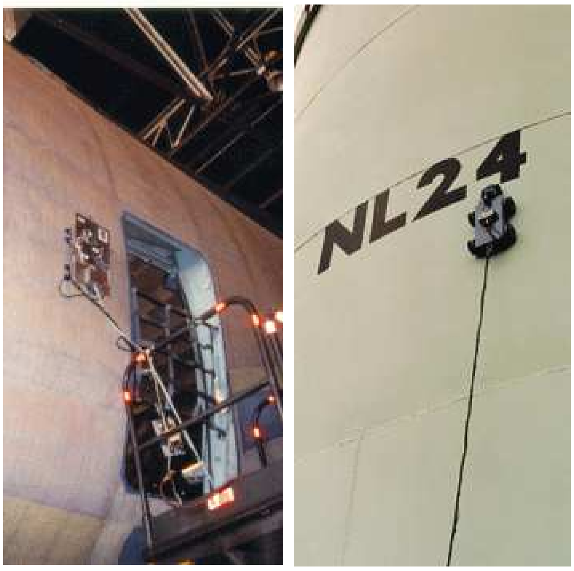


Figure 2.1: Left - MACS [28] developed by the Jet Propulsion Laboratory using vacuum cups for adhesion
Right - Scorpion DCP developed by Silverwing Technologies [26] using magnetic wheels for adhesion

for surface adhesion. It has two legs for linear motion and a rotational element for turning. There is an umbilical cord for power and communications.

Clarifying Technologies [34] have overcome the traditional limitations of vacuum cups with a technology named Vortex Regenerative Air Movement (VRAM) [35, 36]. This technology uses a rotor to create a rotating column of air which has an interior pressure much lower than ambient; this low pressure region causes an attraction force. Note that captive air is recirculated instead of passing through the device. Clarifying Technologies have fitted this into a small battery powered, four wheeled robot (20 × 22 × 11 cm) that is capable of carrying at 0.45kg load over relatively rough surfaces (such as bricked walls). A rechargeable battery provides up to 60 minutes of run time. Figure 2.2 shows the robot with an integrated camera payload.

2.1.1 Miniature Robotic Vehicles

Using off-the-shelf components, functional miniature robotic vehicles can be created as small as one cubic inch [37], an example of this is the Alice Sugar Cube Robot which is shown in Figure 2.3. There is an entire field of robotics dedicated to the design of miniature vehicles with established guidelines on the effects of scaling [38]. Caprari et al. note that smaller vehicles can surmount relatively tall objects,

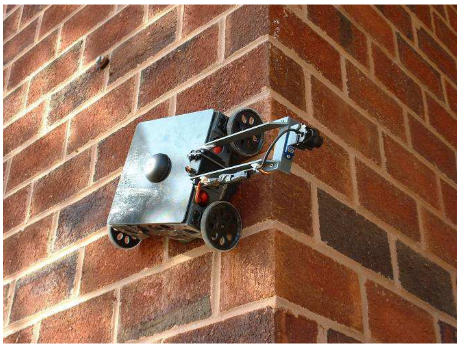


Figure 2.2: Clarifying technologies - Climber III, using an enhancement to vacuum traction technology

since for a cubic robot the energy required to scale the object reduces by $Length^4$, whereas their mass reduces by $Length^3$. They also point out the unfavourable scaling of electromagnetic motors where the torque reduces by $Length^5$ for constant efficiency and they also comment on the lack of availability of high performance motors, sensors and power sources for miniature robots. In the future ultrasonic/piezoelectric motors [39] may provide a favourable alternative to electromagnetic motors for miniature vehicles due to their scalability, but they currently require complicated drive electronics that make the overall actuator larger than electromagnetic alternatives. Robots that are significantly smaller than a cubic inch require MEMS or nano-technology based designs. These are still emerging technologies that require extremely complex manufacture and the results and are not yet used outwith a controlled lab environment [40].

Wilcox et all [41] states that there are optimum robot dimensions for a given terrain that maximises the distance a robot can go before it encounters a non-traversable obstacle. This leads to the notion of an optimum size for a given application rather than the notion that smaller is necessarily better, specifically if a sizable payload has to be carried.

2.1.2 Collaborative Teams of Robots

Splitting an inspection task over a fleet of miniature vehicles has several advantages:

- The individual robots are smaller and lighter, providing better access.
- The individual robots can be less expensive.
- The inspection can be run simultaneously by several robots.

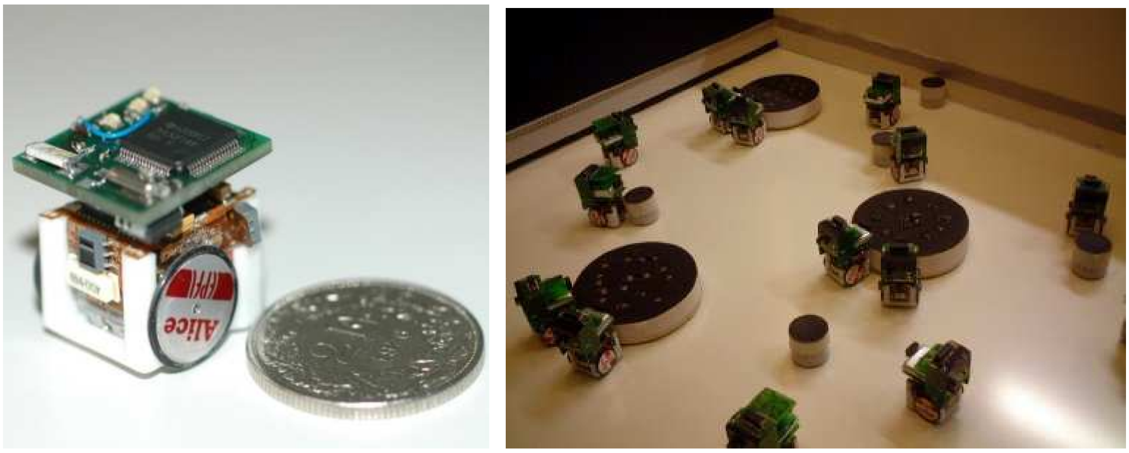


Figure 2.3: Alice the sugarcube robot (2cm x 2cm x 2cm)

- Distributed sensing improves reliability [42] since a single fault only affects one robot and not the whole system.

- The inspection system can be scaled to the size of the inspection by adding additional robots.

However splitting the inspection over a distributed fleet is more academically challenging and has been the source of significant research in the robotics field. Some aspects of this will now be discussed.

Millibots [43] are one of the earliest examples of an effective team of coordinated robots. These devices are used for mapping and exploration using an occupancy grid, Bayesian mapping algorithm [11]. Each Millibot (Figure 2.4) is approximately $7 \times 7 \times 7$cm and carries ultrasonic range finding and miniature cameras to map their environment. The Millibots use a clever system to self-localise. The team is split into two groups, while one group moves the other remains stationary acting as positional beacons, using the ultrasonic range finders for inter-robot distance measurements. This obviates the need for a global positioning system. The approach was able to estimate robot position with an accuracy of 3cm [44]. The team uses a hierarchical command structure with one robot being assigned as team-leader.

The Khepera robot [45] is a popular choice for research into collective robotics [46, 47, 48]. It is disc shaped with a diameter of 13cm and height of 7cm (see Figure 2.4). It has differential drive mechanics. Version III contains a 400MHz Linux module [45] for significant on-board processing. The design is modular allowing the addition of various extensions for wireless communication, sensing, and actuation. Infrared sensors are used for short range obstacle detection and ultrasonic sensors for long range object detection. The low ground clearance of 3mm means that Khepera robots are only suitable for smooth flat surfaces. Antonelli et al. [49] demonstrate a team of Khepera robots on entrapment/escorting missions where the robots continually encircle a target, even when the target is in motion.

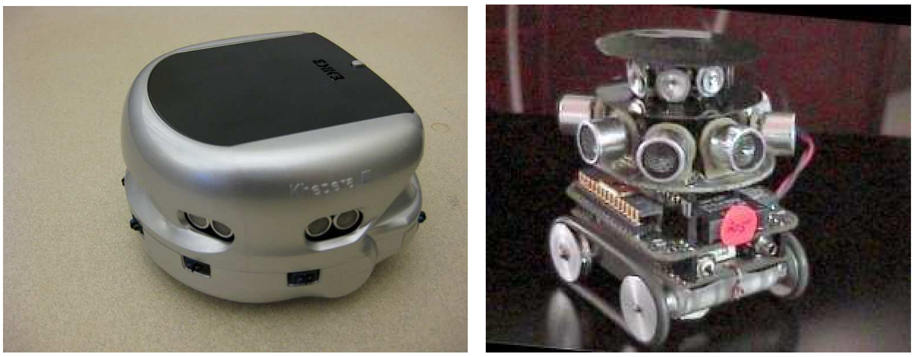


Figure 2.4: Left - Khepera Robot from K-Team [45]
Right - Millibot developed at Carnegie Mellon University

Swarm robotics is the application of swarm intelligence [50], typically found in insects such as ants, to robotics applications. There is typically no central controller and each robot must have enough intelligence to react to the environment and perform actions that contribute to the swarm. The Swarm-bots [51, 52] project that ran until 2005 demonstrated collective behaviour of several identical robots that could link together to move large objects or cross voids wider than a single robot. Each robot was an autonomous robot that could move about, allowing autonomous self assembly of the swarm. The project has made valuable contributions in the fields of distributed control and communication algorithms [53, 54]. The work is continuing under the new project called Swarmanoid [55] which looks to move to a heterogeneous team of robots comprising three distinct classes; Foot-bots which are based on the Swarm-bots robots, immobile Hand-bots with two robotic arms that can be moved by Foot-bots and Eye-bots that are flying robots that carry a camera system to provide a birds-eye-view.

The newest generation of Millibots, the MilliBot train module [56] are also capable of linking together, this time in linear trains. Each train module is fitted with a powerful actuator that allows it to lift several attached modules in a snake like fashion, allowing the train to climb obstacles such as stairs.

In order to locate all defects in a structure, a robotic system must be capable of complete coverage. There are two approaches to this problem, firstly the deliberate

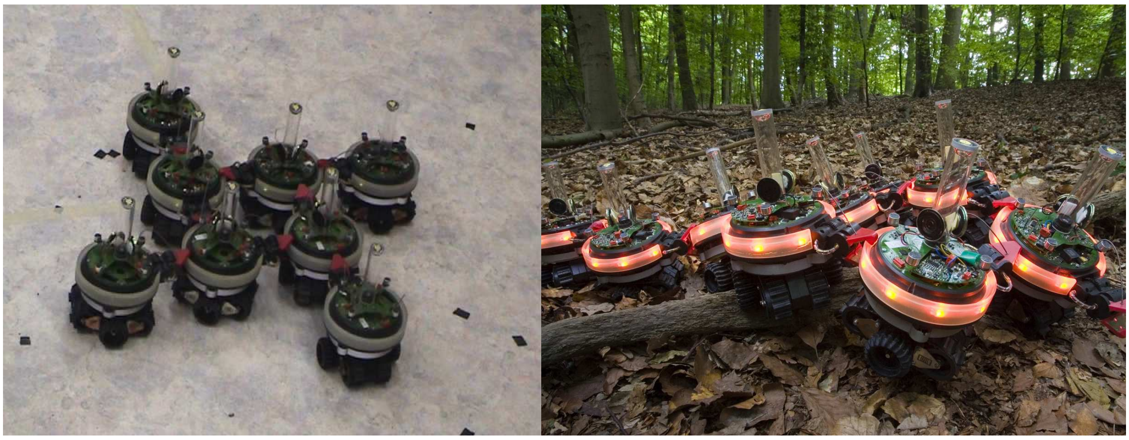


Figure 2.5: Two swarmbots. The left hand picture shows autonomous self-assembly, the right hand picture shows a swarmbot crossing an obstacle

approach in which an accurate positioning methodology can be used in combination with fine vehicle control to execute a search pattern. By constructing a search pattern that covers the entire area, 100% coverage can be assured. The alternative approach is taken from the field of swarm robotics and uses simpler reactive algorithms, such as those developed by the University of the West of England [57]. In this approach, a reactive algorithm runs on each of a number of robots. The algorithm uses a very basic set of rules from which intelligence emerges. By making the search time long enough, adequate coverage can be achieved. This is demonstrated in NDE by Correll et al. [17, 18] who use the miniature robot Alice discussed in Section 2.1.1 to investigate inspection of jet engines. The authors simplify the problem by constructing a 2D approximation of a jet engine (a flat surface with an arrangement of blades) and apply colour markers to allow the robots to identify specific blades. They find that reactive algorithms are very effective in a platform with limited capabilities. The advantage of the reactive approach is that the robots do not require comprehensive positioning or control systems, but this is at the expense of longer inspection times. Additionally the robots cannot provide positional information about the defect, limiting their usefulness. Finally, the problem of differentiating between defects and structural features such as edges is made more difficult by the lack of positional information. For these reasons, reactive algorithms have not been considered in this work.

2.1.3 Robot Positioning

Solving the **S**imultaneous **L**ocalisation **A**nd **M**apping (**SLAM**) problem is considered one of the main achievements in the robotics community in the last decade [58]. SLAM allows robots to dynamically build a map of an unknown location and position themselves relative to the map. The Stanford Artificial Intelligence Laboratory, the Australian Centre for Field Robotics and Oxford Mobile Robotics Group have made notable contributions to the development and realisation of SLAM [59, 60, 61, 62, 63]. Figure 2.6 shows an example map generated by a mobile robot. Robot positioning is considered in more detail in Chapter 3.

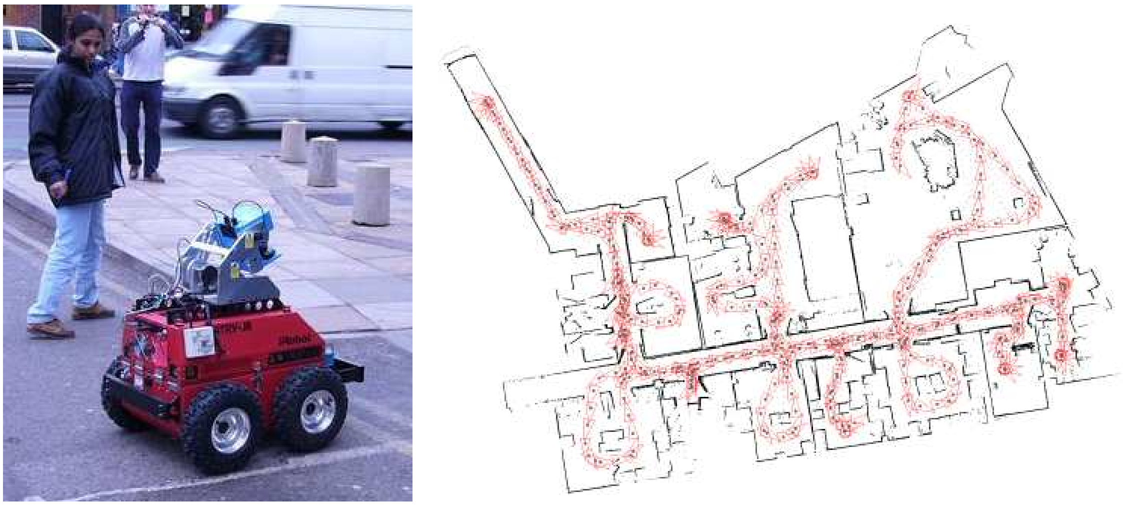


Figure 2.6: Left: SLAM being used in outdoor 3D environment.
Right: 2D SLAM map of Oxford University's Information Engineering Building

Having considered the field of miniature robotics, this Chapter will now focus on the creation of the RSA development robot that formed a crucial part of the project.

2.2 Design Concept

Suitability for ongoing research and system optimisation place contradictory requirements on the design. Developing software on a system with minimal redundancy is laborious and inefficient. When creating a new algorithm for NDE defect detection, it is useful to monitor the raw sensor data, intermediary algorithm calculations and the final algorithm output. However the final design will act as a black box that does not require the additional memory, communications and input/output load required to show the intermediary stages. A development robot was required that featured extensive hardware capabilities that would facilitate effective research, rather than an optimum solution to the short term PhD goals. The concept of a development robot makes a trade off between further miniaturisation or improved power efficiency in return for extensive computational and communication capabilities and greater flexibility.

2.3 The Optimum Size of an RSA

Wilcox et al [41] states that there are optimum robot dimensions for a given terrain that maximises the distance a robot can go before it encounters a non-traversable obstacle. In the case of NDE inspection with miniature robotic vehicles, the ideal dimensions are application specific. The optimum dimensions for the inspection of narrow diameter pipework are very different to optimum dimensions for the inspection a ship's hull.

M. Friedrich created the first generation miniature robot [8] with a volume of 100 × 100 × 100mm. These dimensions are a good trade-off between robot accessibility to tight spaces and general convenience. They are small enough to demonstrate the concept of a team of miniature robotic vehicles in several applications including the inspection of storage tanks, 200mm+ diameter pipework and structural components. A robot of these dimensions is large enough to carry most NDE payloads including eddy current [8], EMATs , ultrasonic (Chapter 4), magnetic flux leakage [8] and visual (Section 2.11.3). Additionally, creating a robot of these dimensions does not need specialist manufacturing techniques such as MEMS and can incorporate enough mechanical power and computational/communication resources to be useful.

2.4 Mechanical Design

The mechanical design is a direct progression from the work of M. Friedrich; engineering drawings can be found in the appendix of his thesis [8]. The chassis is shown in Figure 2.7.

The robot is a two wheeled differential drive vehicle with a rear jockey wheel for stability. The wheels are made out of disc magnets (NeFeB) allowing traction along ferromagnetic surfaces such as carbon steel. Each motor is a 12V DC motor with embedded gearbox with a 207:1 ratio which provides 24 rpm facilitating a top speed of 5cm/s over level surfaces and 3.7 cm/s up vertical surfaces. The chassis was hand machined from aluminium sections which were bolted together. Each of the drive wheels has a 300 count quadrature wheel encoder to monitor rotation. This is discussed in Section 2.9.1.

A collaboration with an undergraduate group project from the Department of **Design**, **Manufacture** and **Engineering Management** (**DMEM**) at Strathclyde University has led to a series of design improvements, these are now briefly discussed. Many of these improvements were made possible using a Rapid Prototyping Machine (Objet Eden 350 [64]). This technology facilities '3D printing' of several plastics and rubber materials and is achieved by building up the structure in fine layers (0.01mm). Each new layer is constructed from powdered material, which is fused into place using a laser. This technology facilitates small volume manufacture of parts with similar quality to injection molding.

- The wheel diameter has been increased from 42mm to 54mm to increase the ground clearance from 5.5mm to 12.9mm.
- Soft rubber tyres have been fitted to the magnetic wheels minimising wheel slip on non-magnetic surfaces.
- In the original design, the rear jockey wheel was manufactured in-house and consistent of several components, this lead to poor tolerances. The assembly had significant friction in both axes of rotation (turning and swivelling) and caused the rear of the robot to move vertically depending on the swivel angle of the wheel. This caused misalignment of the ultrasonic payload discussed in Chapter 4. The jockey wheel was replaced with a steel ball socket assembly which reduced friction and removed any vertical motion. A ring magnet has been mounted above this to direct the magnetic field through the assembly

and into the surface, maintaining magnetic traction. Bearings ensure very little friction on the roller ball.

- The hand-machined aluminium chassis has been replaced with a rapid prototyped ABS plastic chassis which is lighter and can be manufactured to a tighter tolerance. The chassis was simplified to two structural components, the base and the top cover.
- The wheel encoders have been encased to protect them from damage.
- In the original design the motors were attached to the chassis and a shaft extender was fitted to the gearbox shaft. The wheels and optical encoders were attached to this shaft, which was unsupported by the chassis. This put a significant load on the gearbox shaft when the robot was detached from a magnetic surface, which over time, misaligned the shaft, causing components to rub. The revised design uses a bearing on each side of the wheel so that the drive shaft is supported by the chassis.
- The revised robot is now splash proof, with an IP rating of 4.

Figures 2.8, 2.9 & 2.10 demonstrate the new assembly which weighs 610g, almost twice the weight of the original. The weight increase is mostly due to the rugged case that fully encapsulates the robot, the larger wheels and ball bearing rear wheel.

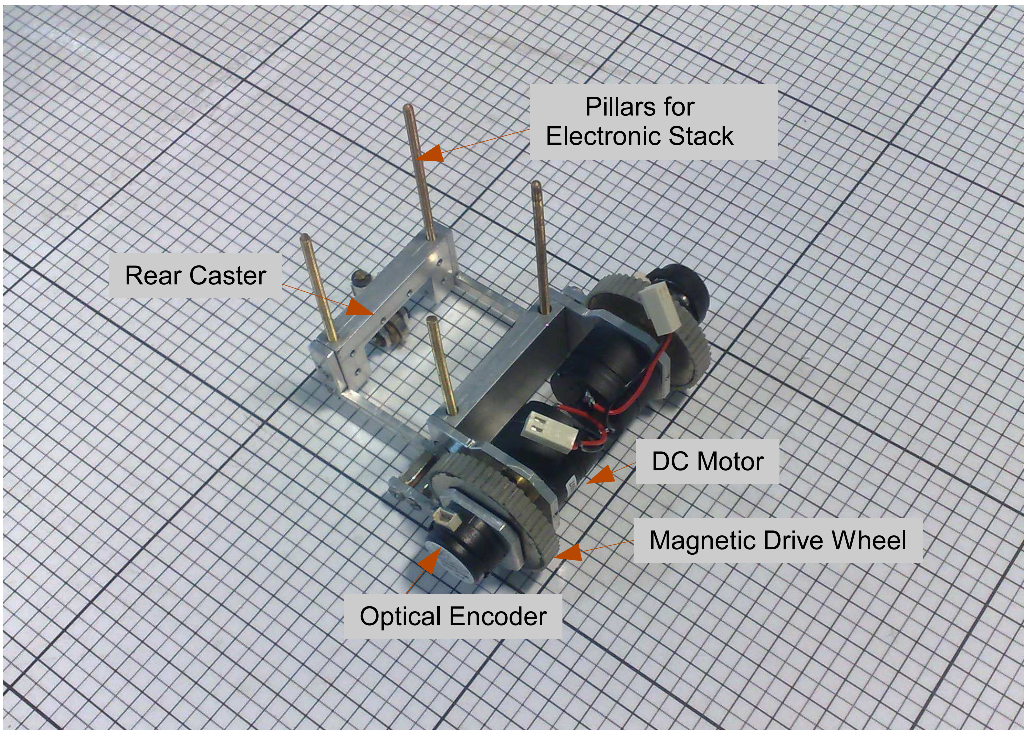


Figure 2.7: Robot Chassis designed by M. Friedrich as part of his PhD [8]. The grid has a resolution of 5 × $5mm$.

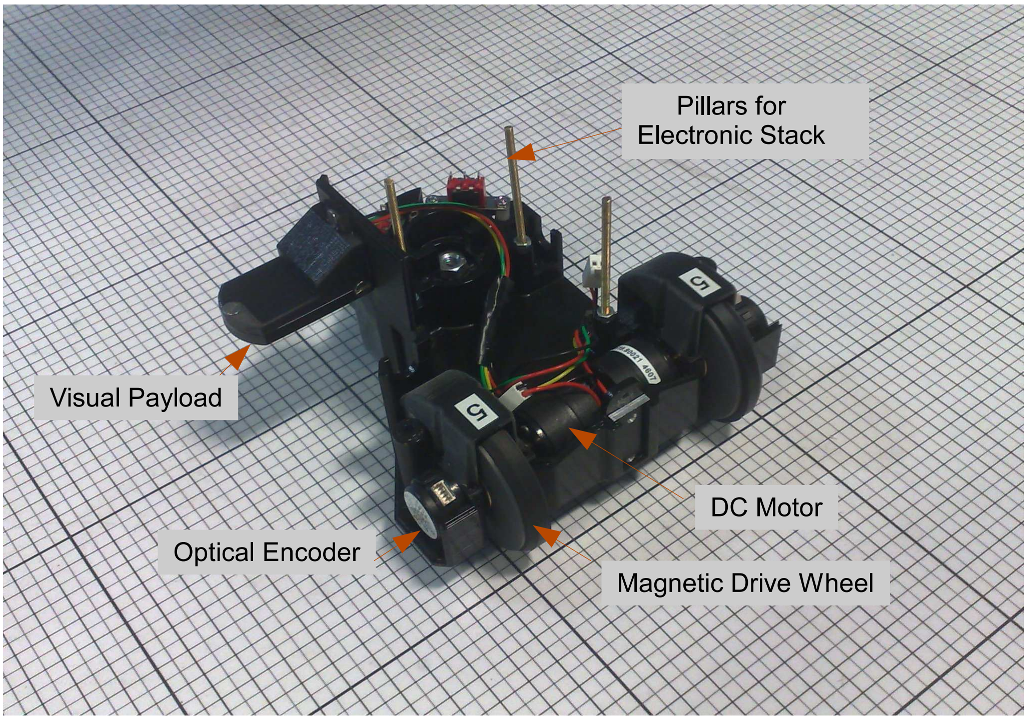


Figure 2.8: Miniature robotic vehicle created using the rapid prototyping machine. The case is not shown. The grid has a resolution of 5 × $5mm$.

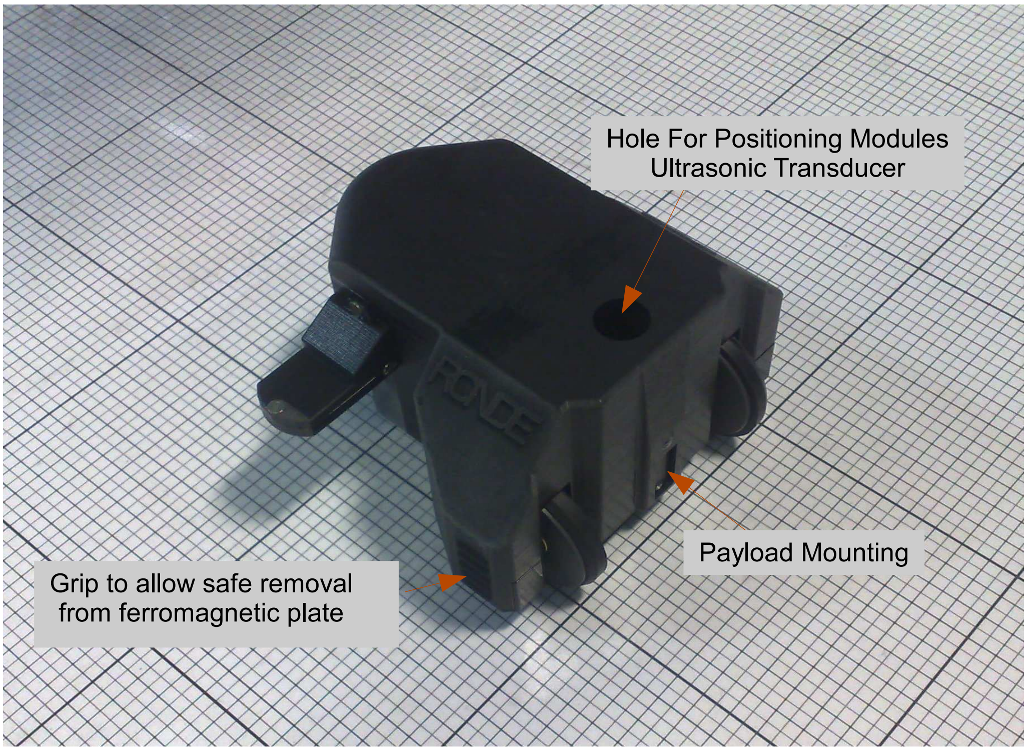

Hole For Positioning Modules
Ultrasonic Transducer
Payload Mounting
Grip to allow safe removal from ferromagnetic plate
RONDE

Figure 2.9: Miniature robotic vehicle created using the rapid prototyping machine. The top cover is shown. The grid has a resolution of 5 × $5mm$.

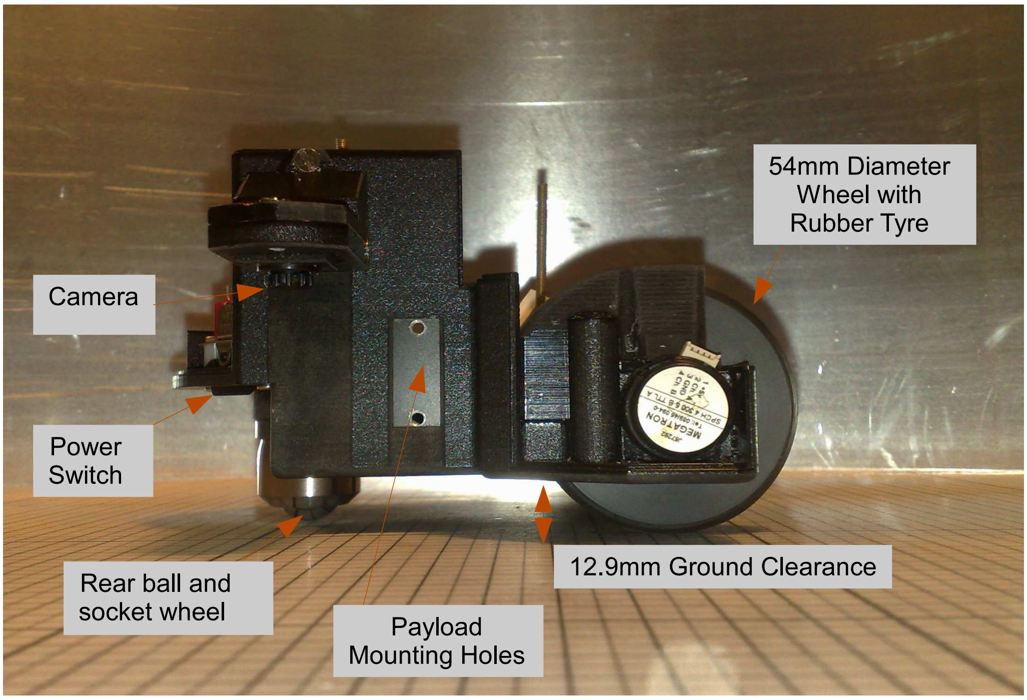


Figure 2.10: Miniature robotic vehicle created using the rapid prototyping machine. View from side; case is not shown. The grid has a resolution of 5 × $5mm$.

2.5 Microcontroller Selection

The focus will now shift from the robot's mechanics to the design of the electronics.

The robot carries a Gumstix [65] Connex 400xm embedded computer for high level processing tasks such as robot positioning, image processing and data fusion. The Connex 400xm measures 80mm x 20mm and runs a stripped down Linux kernel on an ARM7 processor. The PXA255 CPU runs at 400MHz and the system has 16MB of Flash memory, 64MB of RAM with extensive communication capabilities in the form of UARTs, an I2C controller and SPI support. **S**ecure **SH**ell (**SSH**) is used to remotely access the computer giving full access to the file system and the ability to run programs interactively. The 'Robostix' peripheral also from Gumstix [65] was used to add an 'atmega128' 8 bit microcontroller providing a 10bit analogue to digital convertor with eight channels, several timers and multiple I/O ports. A 400 kbit/s I2C bus interconnects the two processors. The microcontroller provides a low-level realtime hardware interface for tasks such as closed loop control of the motors - Section 2.9.3. Splitting on-board computation over the two processors keeps simple real-time tasks running effectively on the microcontroller while maintaining the flexibility of an embedded computer with full operating system for higher level tasks.

2.6 Communication Protocol Selection

The IEEE 802.11g (Wi-Fi) digital communication protocol was selected for the RSA development platform. It provides high bandwidth (54Mbps) and a reasonably long range (100m line of sight). Wi-Fi can be configured to act as an ad-hoc network that requires no base station and dynamically changes to encompass all devices in range. This makes Wi-Fi ideal for a dynamically changing team of robots. In the longer term, the system may migrate to Zigbee, which also provides a network protocol, but trades off lower bandwidth to provide a lower power solution.

2.7 Circuit Board Design

The electronic design of the development robot is based on the concept of a stack of printed circuit boards interconnected by a 40 pin bus. The bus minimises wire connections between boards providing modularity whilst maintaining robustness. New boards can be added to the stack to extend the robot's capabilities making the arrangement ideal for research. Since the signals go through the bus on every level the order and number of the boards is insignificant. The bus is realised using PC/104 2.54mm pitch connectors which is extremely convenient to work with. The connector has the same pitch as a conventional breadboard and is large enough for reliable in-house manufacturing, allowing prototype sensor payload layers to be rapidly reiterated. There are much smaller 0.5mm pitch surface mount alternatives, but convenience was more important than miniaturisation at this stage.

2.8 Electronic System Design

The core development robot contains the mechanical hardware, a battery and two electrical layers. The first electrical layer provides a hardware interface and power management; it is discussed in Section 2.8.1. Above this a processing layer houses the embedded Linux computer, microcontroller and Wi-Fi module. This is discussed in Section 2.8.2. A system diagram is shown in Figure 2.11 and a photograph is shown in Figure 2.12. An optional third layer provides NDE payload electronics such as the ultrasonic payload discussed in Chapter 4. A miniature camera module can be attached for visual inspection; this is detailed in Section 2.11.3. Finally, a global positioning module can be attached to improve positional accuracy, this is discussed in Section 2.11.1.

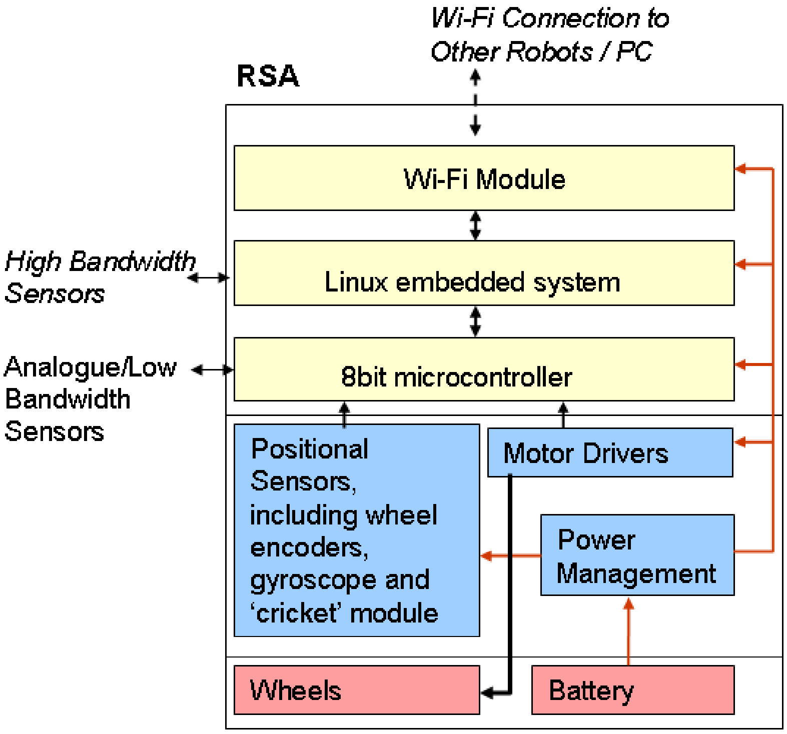


Figure 2.11: System diagram of the development robot

2.8.1 Motherboard

The motherboard is shown in Figure 2.13, with the schematic in Figure A.1, Appendix A. The sockets have been chosen to ensure that plugs cannot be inserted in the wrong place or with the wrong orientation. This minimises the risk of damage by errors such as power being attached with the wrong polarity. A reverse battery protection circuit on the board acts as a second line of defence. The board provides connectors to interface with the wheel encoders which track wheel rotation and embeds an ADXRS150 [66] 150 deg/s gyroscope which tracks angular acceleration as the robot turns. An L293D [67] motor driver is used to control the two drive motors, it is similar to a conventional H-Bridge [68] motor driving circuit, mounted on a single IC. Using control logic from the processing board (routed through the bus), the motor driver can set the direction and speed of the two drive wheels. The speed is controlled by **P**ulse **W**idth **M**odulation (**PWM**), which is a relatively high frequency square wave signal with controllable duty cycle. The motor sees the running average of signal amplitude, so a signal with a duty cycle of 50% is equivalent to a DC signal at 50% amplitude of the PWM high level, i.e. 50% duty cycle 5V PWM $\simeq$ 2.5V DC. The motor driver applies the 12V battery voltage over the motors tracking the drive signal's PWM. The wheel direction control inputs are shown in Table 2.1. It is possible to save digital output pins by configuring channel B as the inverse of channel A, but was emitted since it requires an additional inverting component. The L293D cannot be damaged by an combination of inputs, unlike a conventional H-Bridge which shorts with directional inputs of '0,0' or '1,1'. The L293D driver simply routes the same voltage to each motor terminal, removing the voltage drop over the motor. A 5V LM2825N-5.0 [69] switching mode regulator reduces the battery voltage to a regulated 5V at up to 1A with an efficiency of 80%. Although 1A is significantly more than typically required, it is in keeping with the development robot concept. A red 'low battery' LED is configured to come on when the battery voltage drops to 9V. A scaled version of the battery voltage is routed to the signal bus so that it can be converted to a digital format and monitored by the robot's software. The switch configures the circuit into one of three modes. 1. battery supply, 2. external supply and 3. battery charge mode. In battery charge mode the external supply is routed to the battery, facilitating recharging. The motherboard was designed to be a stand alone board, however if it can be integrated into a robot with case mounted connectors and switches, by simply omitting the on-board equivalents. This was the case for the robot discussed in Section 2.4.

Channel A	**Channel B**	**Result**
0	0	Stopped
0	1	Left
1	0	Right
1	1	Stopped

Table 2.1: Wheel directional control using two binary inputs

Battery Technology	Gravimetric Energy Density (Wh/kg)
Nickel-cadmium	45-80
Nickel-metal-hydride	60-120
Lead acid	30-50
Lithium-ion (cobalt)/Lithium-Polymer	150-190
Lithium-ion (manganese)	100-135
Lithium-ion (phosphate)	90-120

Table 2.2: Comparison of rechargeable battery technologies [70]

2.8.2 Processing Board

The processing board is shown in Figure 2.14 and the schematic is shown in Figure A.2, Appendix A. This board interfaces the embedded computer and microcontroller discussed previously with the 40 pin bus, powering the two processors and providing access to the required I/O ports (analogue and digital).

2.8.3 Battery

As summarised in Table 2.2, Lithium Polymer batteries currently offer the highest gravimetric energy density of all rechargeable battery technologies [8][70].
A 11.1V 3 cell (3 x 3.7v) Lithium-Polymer battery with dimensions of 40 × 30 × 20 mm provides the robot with 980mAh. The core RSA draws 250mA from the battery which facilitates approximately four hours of operational time from a single charge. There is extensive scope to extend this by reducing the processor clock and communication systems transmitter power when full capacity is not required.

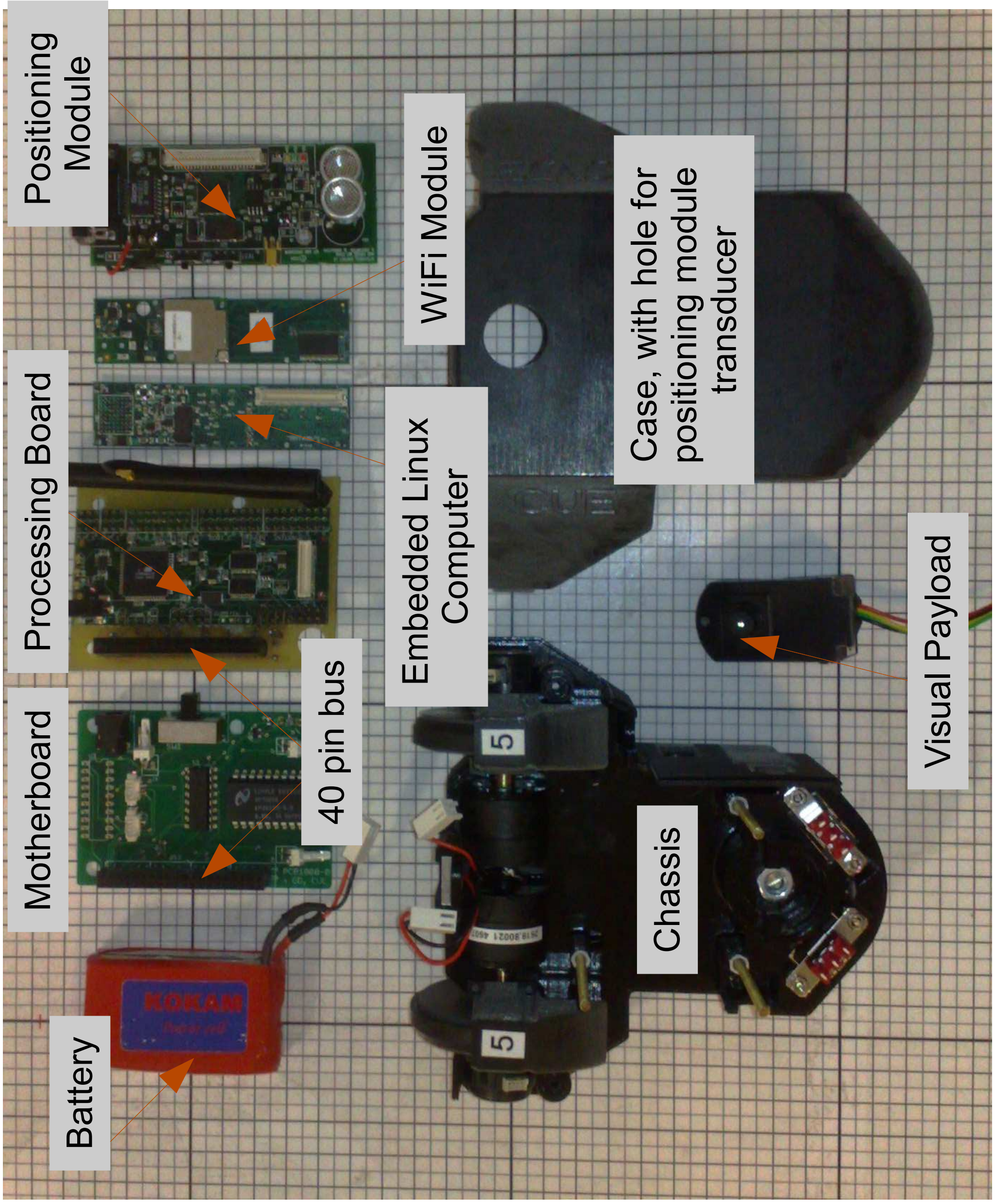


Figure 2.12: Photograph of a disassembled robot

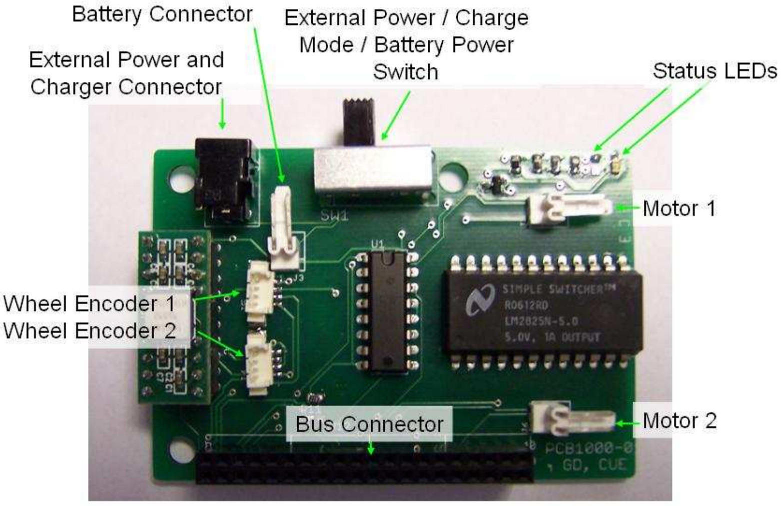

Battery Connector
External Power / Charge Mode / Battery Power Switch
External Power and Charger Connector
Status LEDs
Motor 1
Wheel Encoder 1
Wheel Encoder 2
Bus Connector
Motor 2

Figure 2.13: Motherboard PCB

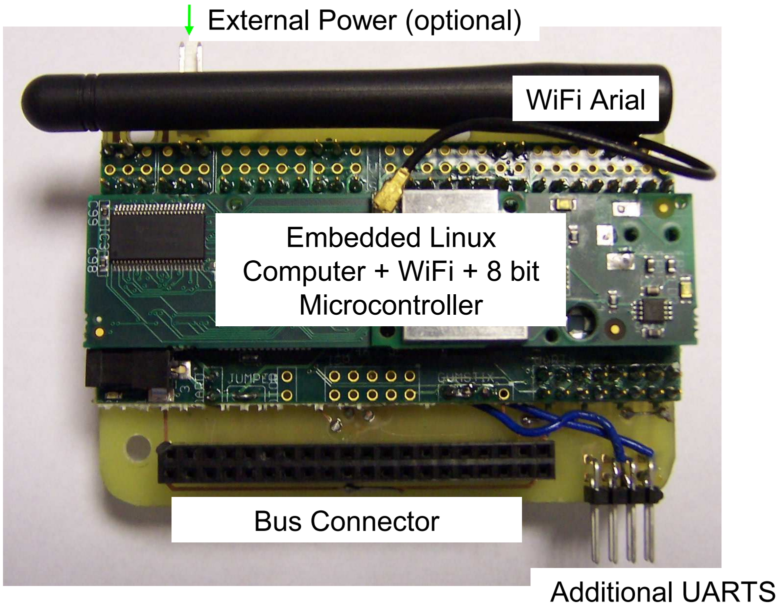

External Power (optional)
WiFi Arial
Embedded Linux Computer + WiFi + 8 bit Microcontroller
Bus Connector
Additional UARTS

Figure 2.14: Processing board PCB

2.9 Robot Control

2.9.1 Calculating Wheel Velocity using a Wheel Encoder

The displacement and speed of each wheel is calculated from the encoders mounted on the drive shaft. These quadrature encoders have two output channels (A and B) which are 90^o out of phase, as shown in Figure 2.15. By comparing the channels A and B, a phase value can be calculated that goes from 1 - 4 and corresponds to encoder logic levels of '0-0', '0-1', '1-0' and '1-1' on channels A and B respectively. Summing the relative phase changes produces a count, which increments for clockwise rotation and decrements for anticlockwise rotation. Since the count is updated on a change of phase it has a resolution four times finer than either channel A or B.

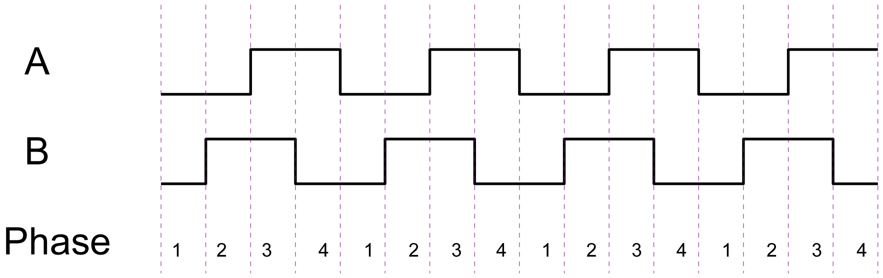


Figure 2.15: Quadrature Encoders

If there is no wheel slip, the tangential distance travelled by the wheel can be calculated using Equation 2.1.

$$distance = nCounts \times \frac{wheel\ circumference}{4 \times encoder\ resolution} \tag{2.1}$$

In this case, the encoder resolution is 300 and the wheel circumference is $\pi \times 54mm$. The rotational speed of the wheel is calculated using the time taken for the wheel to rotate a set number of counts, i.e.

$$speed = \frac{D}{t_N} \tag{2.2}$$

Where t_N is the time it took for the wheel to rotate N counts. D is a constant used to scale the speed. For SI units:

$$D = \frac{N}{4 \times encoder\ resolution} \times wheel\ circumference \tag{2.3}$$

Alternatively D can be chosen to scale the speed between 0 and 100%. The time in milliseconds is generated by one of the microcontroller's timers.

An implementation issue occurs when the wheel is brought to rest. Since the speed is only updated on new encoder readings the speed reaches a minimum value calculated at the last time the wheel rotated N counts. To correct this, the time since the last update is recorded $t_{LastUpdate}$. If $t_{LastUpdate}$ exceeds the previous value of t_N the speed is updated using $t_{LastUpdate}$ rather than t_N in Equation 2.2. In this way the wheel speed gradually decreases. In practice a threshold is set for $t_{LastUpdate}$, above which the wheel speed is set to zero.

2.9.2 Robot Motor Calibration

The robot drive wheels are driven by pulse width modulation (PWM). The frequency of the PWM signal is sufficiently high that the motor sees the average of the signal. As the duty cycle of the pulse is increased, the power to the wheel increases thus increasing the velocity. The measured velocity of the wheel does not vary linearly with duty cycle, but this can be linearised using piecewise linear calibration. Figure 2.16 shows the raw output against PWM, Figure 2.17 shows the calibrated wheel outputs, created from 20 linear sections. The calibration procedure can be performed automatically from the GUI. This function drives the robot forward, then backward, each time incrementing the speed by 10% until the full range of speeds has been covered. At each point, the PWM and wheel speeds are recorded and stored in a 'look-up' table. During regular operation, the PWM required for a given speed is read from the look-up table, using interpolation as required.

2.9.3 Wheel Speed Controller

Calibration alone cannot compensate for a disturbance or change in operating conditions (such as moving to an inclined surface). Therefore, the velocity of each drive wheel is controlled by an embedded Proportional Integral and Differential (PID) controller. This is shown in Figure 2.18. The error signal $e(t)$ is calculated as the difference between the desired and measured wheel speeds. The controller action c(t) is derived by considering the magnitude of this error, its accumulation (integral) and its rate of change (derivative). The controller is tuned by altering the effect of each of these components through three gain's K_P, K_I and K_D. The summation between the control signal c(t) and the motor driver makes the system a velocity controller since it controls a modification to the drive signal d(t) rather than the

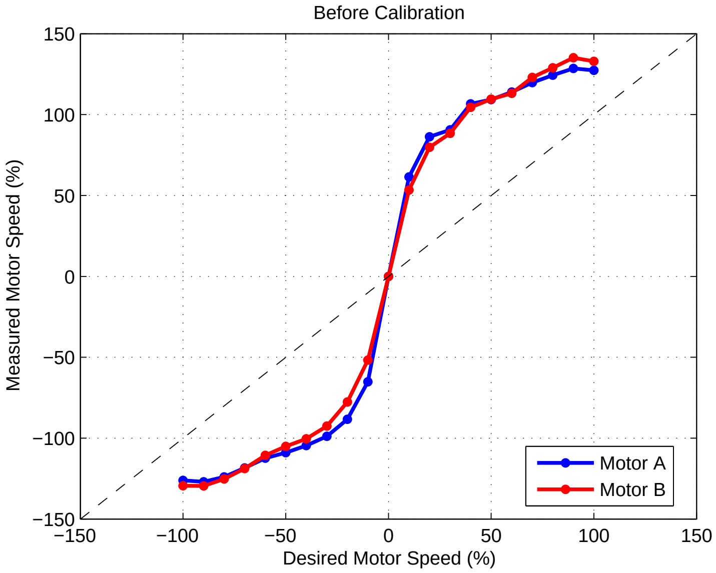


Figure 2.16: Measured wheel speed plotted against PWM, before calibration. The Y axis is a percentage of the desired maximum speed.

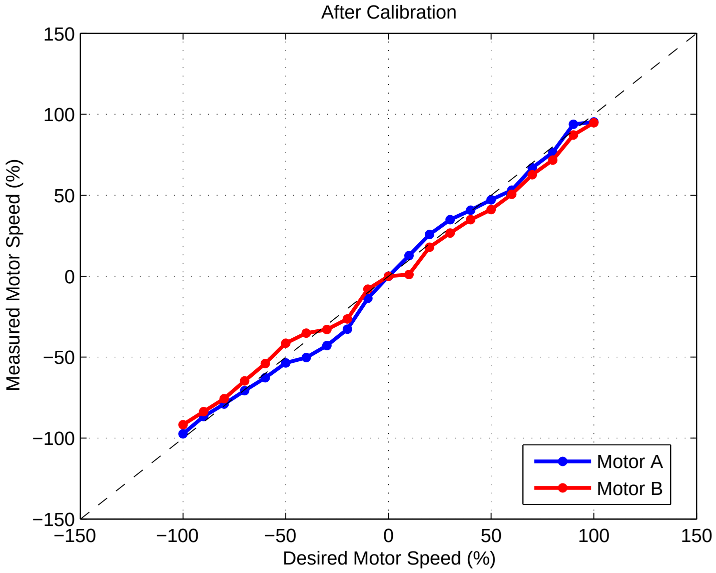


Figure 2.17: Calibrated wheel speed plotted against PWM with calibration. The Y axis is a percentage of the desired maximum speed.

drive signal itself. In this configuration, zero error in the desired speed will leave the drive signal unmodified. An additional parameter is used to set the update rate of the controller, which, for stability, should update at a rate equal or slower than the encoder sample rate. In order to minimise the time taken for the wheels to accelerate to the drive velocity, the control logic uses the calibration data to set the drive signal before enabling the PID controllers to tune and monitor the velocities. This enables rapid acceleration without significant overshoot.

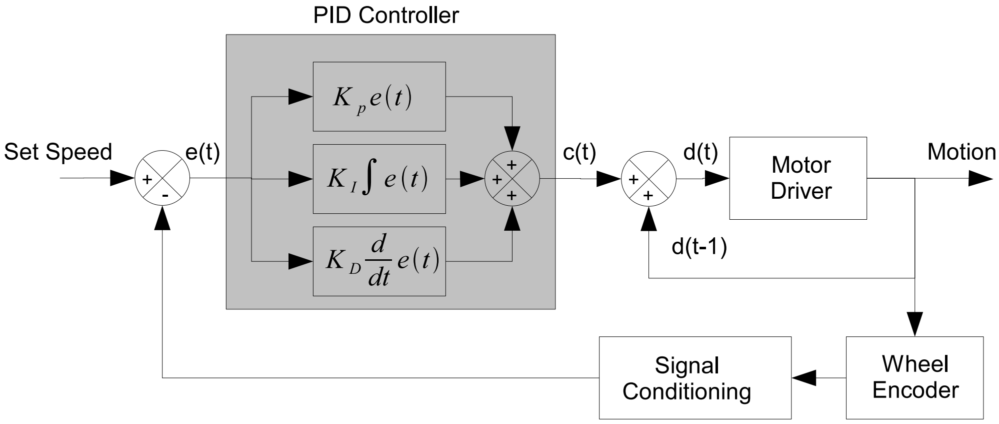


Figure 2.18: Closed loop control of wheel velocity

2.9.4 Path Generation

The robot's path is constructed from a series of 'move to coordinate' commands. For each 'move to coordinate' command, the robot typically turns to face the goal and then travels in a straight line until it arrives. The robot slows down when it approaches the destination to prevent overshoot of the goal coordinates. A 'distance from destination' threshold is defined at the point the robot switches to the slower speed and a second threshold is defined for when the robot is close enough to the destination for the manoeuvre to be considered complete. Table 2.3 shows the four parameters used to control a straight line manoeuvre. The same approach is taken for turning on the spot.

The 'move to coordinate' command is defined as follows:

- Destination coordinates
- Option to initially turn on the spot to face the destination.
- Option to stop motion at the end of the command

Variable	Description
Velocity Fast	Regular velocity for manoeuvre
Velocity Slow	Velocity near destination
Velocity Threshold	Distance at which to switch between velocities, typically 10 - 20 mm
Stop Threshold	Distance at which manoeuvre is complete, this is very short distance, for example 0.5mm.

Table 2.3: Parameters controlling straight line movement

If the option to stop at the end of the command is 'disabled', the robot will remain at full speed right to the end of the command. By disabling the subsequent command's turn, it is possible to make the two commands flow together. Optimum robot path generation is discussed further in Chapter 3 which is concerned with robot positioning and control.

2.9.5 Coordinating Two or More Robots

Since the path of each robot is defined as a series of points, robot paths can be synchronised by inserting a wait command into the path. This causes a pause until all robots are ready, before progressing. Finer synchronisation is achieved by continually monitoring each robot's progress and controlling the drive speeds of each robot to maintain synchronisation. This is achieved by dynamically modifying the set speeds detailed in Table 2.3.

2.10 Software Development

2.10.1 System Structure

The system software has been split into three core components. The microcontroller firmware that runs on the AVRMega128, the program that runs on the robot's embedded computer and the PC server application. The microcontroller firmware is responsible for low level hardware monitoring and control such as closed loop wheel control and battery level monitoring. Like most microcontroller applications, it was written in C. The robot's main application runs on the Linux computer and is the home of most of the embedded algorithms. It communicates with the server through Wi-Fi and controls the robot by sending commands to the firmware. It was written in C++. The PC server has two functions. Firstly, it monitors all robots controlling the host side of the Wi-Fi link to each robot and secondly, it acts as a

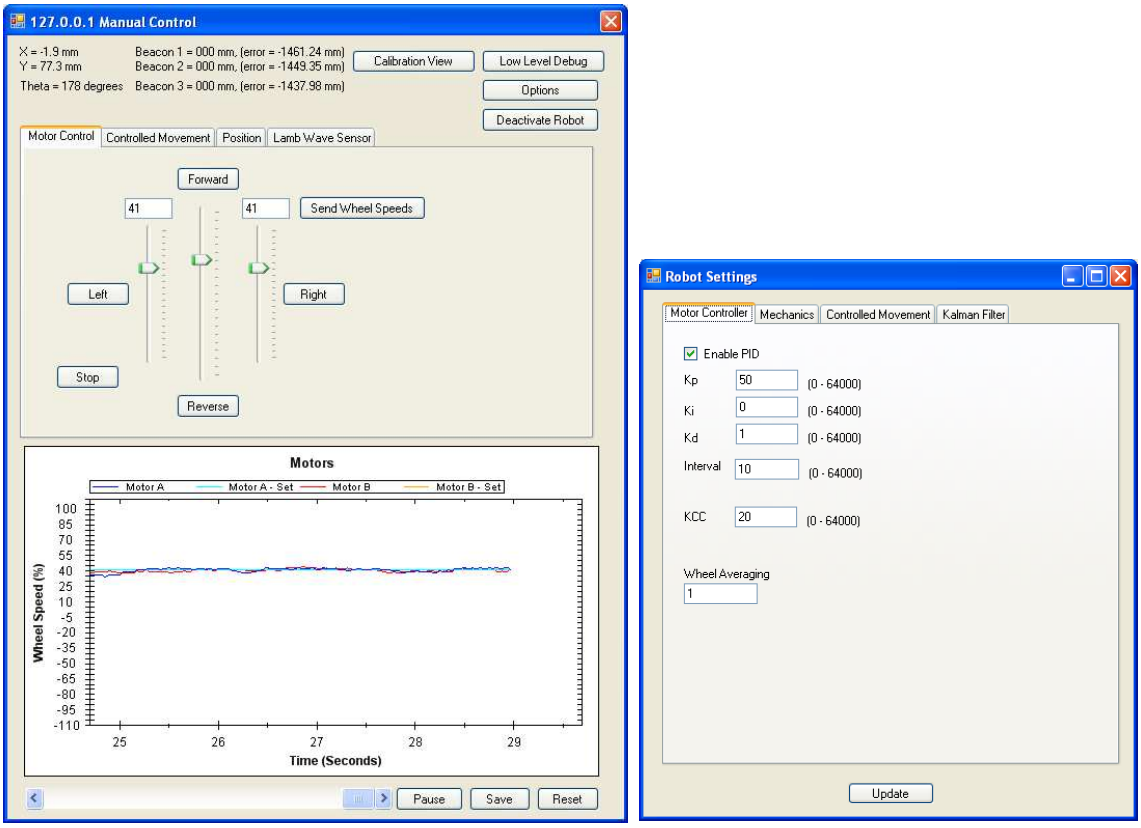
GUI for user control of the robotic fleet. The GUI was written in Microsoft's .NET Framework (v2.0) [71] using the C# language in Visual Studio 2005. The .NET framework provided a substantial set of libraries for standard functionality, such as network access, container types and Windows forms. C#/.NET was integrated with Managed DirectX [72] to create 3D graphics.

2.10.1.1 Robot Server

Berkeley sockets is the standard abstraction for communications over an internet protocol-based network. The .NET sockets library was used to create the PC server. Each robot uses an implementation of the sockets' protocol to create a TCP/IP connection with the server. The network is configured to be ad-hoc with the robot searching for servers with IP addresses ranging from 192.168.74.129 to 192.168.74.150. The robots have addresses 192.168.74.10x where 'x' is the robot number (1-6 for this work). Once the connection is established, strings can easily be sent between the PC and individual robots. On both the PC and robots the communications code is run in a separate thread which receives incoming commands and places them into a buffer. The main processing thread routinely checks this buffer for commands and responds as required. The specific commands include instructions to the robot such as 'move to coordinate' and data back to the PC that provide sensor data and the robot's status.

2.10.1.2 Individual Robot Graphical User Interface

The system is designed so that the robot's parameters are stored on the embedded computer's flash memory (rather than on the PC). Each robot maintains its own personal settings, in a file called 'config.ini'. This file is loaded at boot-up and saved each time a parameter is changed. Since it is often convenient to tune these parameters in real-time, the GUI has a subsection which is essentially a GUI for the individual robot. This can be accessed for any robot connected to the server. Figure 2.19 shows this interface. It allows adjustment of any of the robot's parameters, facilitates manual control of each motor provides and the ability to send basic move commands, such as move forward a specific number of millimeters or turn to a specific angle. A strip chart provides a history of the wheel speeds and set speeds allowing for convenient adjustment of the control parameters. The robot's status (pose, wheel speeds, battery level etc) are buffered and can be saved to file for future evaluation. An additional dialogue is used to perform the calibration procedure detailed in Section 2.9.2. Finally this interface allows adjustment and testing of any payload attached to the robot, for example the ultrasonic payload discussed in Chapter 4.



127.0.0.1 Manual Control

X = -1.9 mm
Y = 77.3 mm
Theta = 178 degrees

Beacon 1 = 000 mm, (error = -1461.24 mm)
Beacon 2 = 000 mm, (error = -1449.35 mm)
Beacon 3 = 000 mm, (error = -1437.98 mm)

41
41

Robot Settings

☑ Enable PID

Kp	50	(0 - 64000)
Ki	0	(0 - 64000)
Kd	1	(0 - 64000)
Interval	10	(0 - 64000)
KCC	20	(0 - 64000)

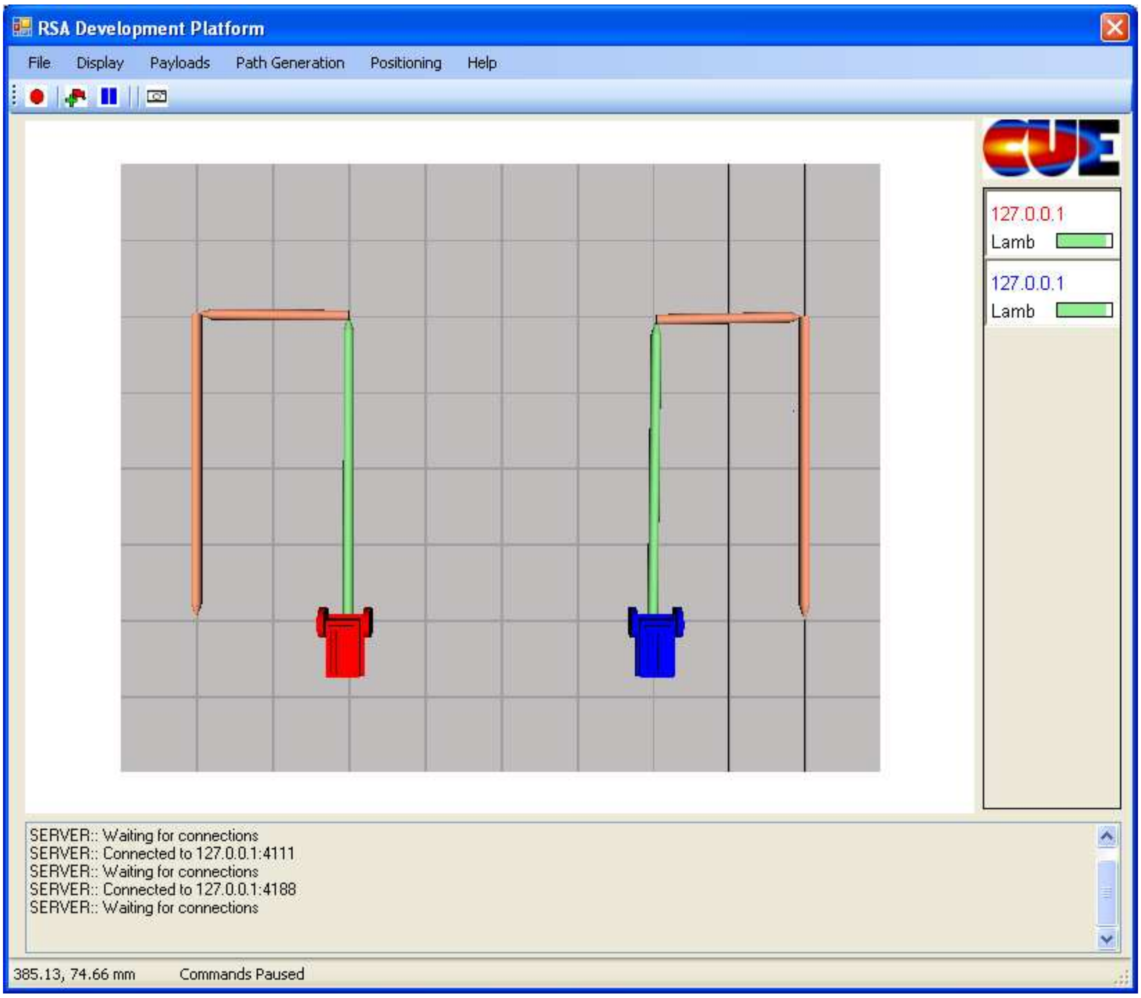
Wheel Averaging
1

Figure 2.19: GUI for individual robots

2.10.2 Graphical User Interface

The GUI is designed around a point and click interface which allows the operator to control a fleet of robots using the computer mouse to interact with virtual robots in a virtual representation of the world. The robots are represented in a 3D world created using Microsoft's DirectX API [72]. An example of two robots on a plate is shown in Figure 2.20; a path has been plotted for the robots. The path changes colour after a 'waypoint'; the robots will synchronise at this point. The robots can be selected by clicking on them, and moved by clicking on the destination. Complex paths can be created by stacking these commands as shown in Figure 2.20. Paths can be saved to file and loaded at a later date. The user can zoom in and out and move the camera to change the viewpoint.

The algorithm to select and move robots requires the transformation of 2D screen coordinates into 3D virtual world coordinates. This was achieved by projecting a ray



```
SERVER:: Waiting for connections
SERVER:: Connected to 127.0.0.1:4111
SERVER:: Waiting for connections
SERVER:: Connected to 127.0.0.1:4188
SERVER:: Waiting for connections
```

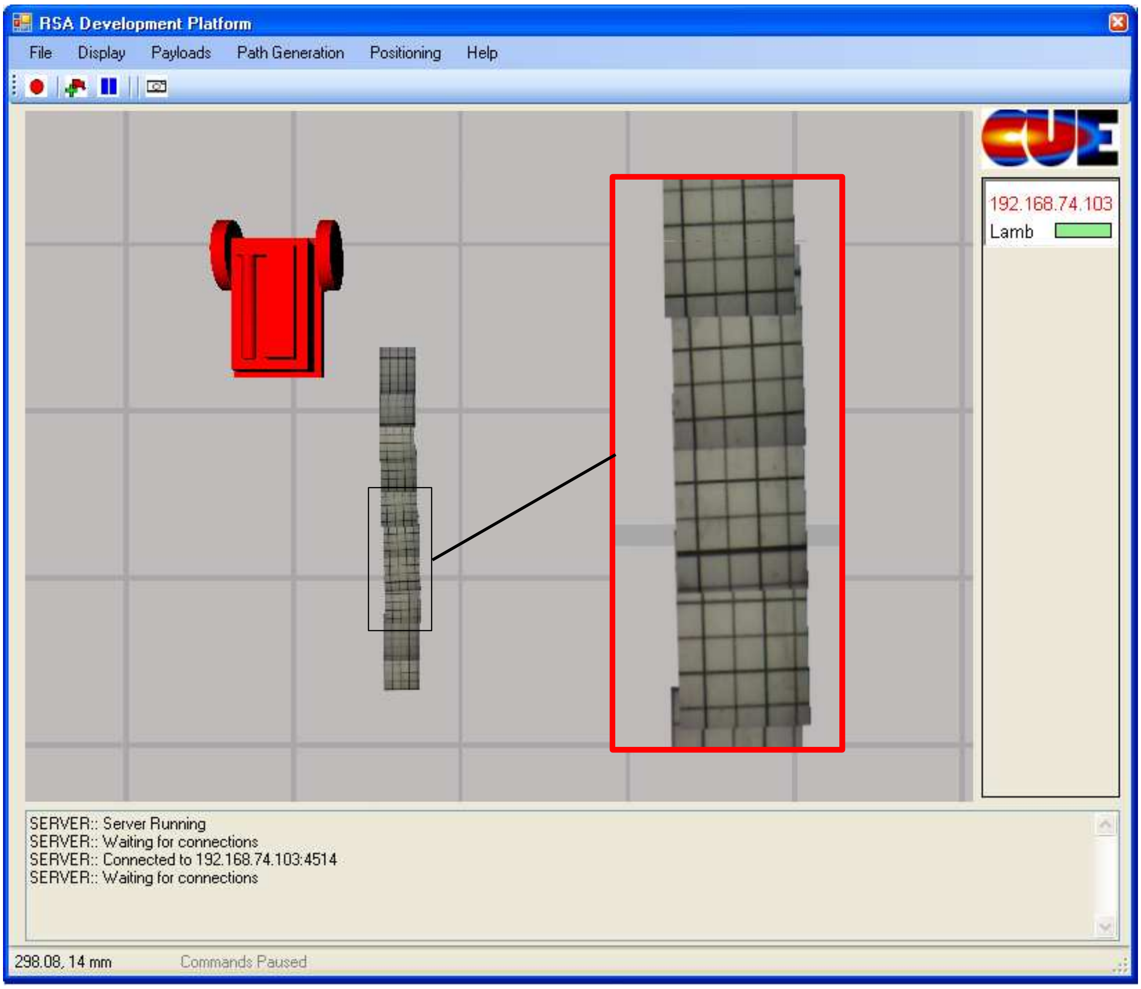
385.13, 74.66 mm   Commands Paused

Figure 2.20: Two robots shown connected to the host program. A path has been plotted for both robots.

forward from the pointer and testing for intersection with either the NDE sample or any of the robots. Intersection between the ray and a robot was used to select a robot. The point of intersection between the ray and NDE sample was used as a destination for robot move commands.

2.10.3 Presentation of NDE results

The 3D representation of the plate is used to show NDE results giving spatial context to each result. For example, when the visual payload takes a picture, it is painted onto the virtual plate at the location/pose it was taken. Taking multiple pictures produces an aggregated high resolution image of the plate surface. An example of this is shown in Figure 2.21. The implementation is as follows.



```
SERVER:: Server Running
SERVER:: Waiting for connections
SERVER:: Connected to 192.168.74.103:4514
SERVER:: Waiting for connections
```

Figure 2.21: Aggregate high resolution plate image shown in context in virtual world. The grid has a resolution of 5mm.

Consider a coordinate system aligned with the robot, with the origin placed between the two drive wheels and the y axis facing in the direction of motion. The robot rotates around the origin of this coordinate system. In this work, the visual payload is mounted at 63, 65 mm. The coordinates of each image must be transformed to the global coordinate system, taking account of the robot's pose. This can be achieved using standard coordinate transforms that are detailed in Appendix B. Performance constraints limited the resolution of the master plate image to 5 pixels/mm, so each new image had to be spatially down sampled to match this resolution (The original high resolution images are retained, but not displayed in the virtual world.) The down-sampled image is then rotated to match the robot's orientation and painted onto the master image at the appropriate coordinates. It is inefficient and hence detrimental to performance, to update the whole NDE plate texture each time a new result is received. Instead, the master plate is broken into tiles and the algorithm searches for tiles that are affected by the new result. Each relevant tile is updated with the portion of the new image that overlaps the tile. The NDE plate image can be exported to file for future use, either as a bitmap, or an ASCII file that can be imported into other software packages such as Matlab [73].

2.10.3.1 Improving the Visual Payload Performance

To achieve optimum image quality the robot needs to be stationary when it captures each picture from the Visual Payload detailed in Section 2.11.3. If the camera was upgraded to a model that allowed rapid exposure, it should be possible to take pictures in motion facilitating faster scans.

The robot's positional inaccuracy can be up to 15mm (Chapter 3) which is a significant proportion of the 30 x 26mm area captured in each image. This can lead to poor alignment of images, especially between successive scan lines. Although future improvements in hardware and algorithms will reduce this inaccuracy, it is unlikely to be reduced to a sub-pixel level (30 mm / 640 pixels $\approx$ 0.05mm per pixel). The elegant approach would be to extract features from the overlapping region between images to improve alignment, ideally feeding this information into the robot positioning algorithm to improve the robot's positional accuracy. The difficulty would be in extracting sufficient features from relatively barren metal surfaces and being able to perform the alignment calculation quickly enough that it would not need to be done in post-processing.

2.10.4 Microcontroller Code

The embedded microcontroller is responsible for interfacing with hardware and performing low-level time critical operations. These include the wheel PID controller (Section 2.9.3), wheel velocity measurement and calibration (Sections 2.9.1 and 2.9.2) and movement to coordinates (Section 2.9.4). The embedded ADC is used to measure the gyroscope and battery voltage. The kinematic model of the robot described in Section 3.3.2 is implemented on the microcontroller so it can track the robot's pose. If a global positioning system is available, the pose is updated routinely by the Kalman Filter (Chapter 3) that runs on the embedded computer to prevent the accumulative error experienced when relying on dead reckoning alone.

A one millisecond timer calls the core function that processes all routine tasks, such as polling the wheel encoders, processing the PID controller and updating the robot pose. A state machine is used to process commands that need to be monitored over a period of time, such as 'move to coordinate' or 'turn to angle' commands. The command is received by the I2C interrupt and put at the end of a command buffer. When the robot is in the 'idle' state it pulls the front command from this buffer and switches the appropriate state, such as the 'moving to coordinate' state. It stays in this state until the manoeuvre is complete at which point it transitions back to the 'idle' state. Several commands can be queued in the command buffer. The progress of the current manoeuvre can be requested if required for inter-robot synchronisation (Section 2.9.5). The ADC is configured to automatically run in a cyclic mode that samples each active channel in turn and puts the results in an array for processing by the main function.

2.11 Peripherals

The following peripherals were integrated into the robot development platform.

2.11.1 Cricket Indoor Positioning Modules

As discussed in Chapter 3 a global positioning system can be used to enhance the positional accuracy of the robots. Building upon the work of M. Friedrich [8] the Cricket Indoor Positioning System [74, 75] was integrated into the system. The Cricket modules are set up in a pitch-catch arrangement. The relative distances between transmit and receive modules is calculated by measuring the time of flight of an ultrasonic burst. The transmit module simultaneously emits a 433MHz RF pulse and a 40KHz ultrasonic pulse. The receiver calculates the time difference between these pulses, assuming the RF pulse is instantaneous. The distance is calculated by multiplying the transit time by the propagation velocity of sound. Equation 2.4 [76] provides this velocity, where θ is the temperature in degrees centigrade. By using three or more fixed beacons and a module mounted on the robot, a trilateration calculation can be performed to calculate the robot position. Only one RF/US transmission can occur at any one time to prevent interference of US signals. However one active beacon can be simultaneously received by several listeners. The active beacons are battery powered, so do not require any connections to the host PC. The listeners return the time of flight through a serial interface which must be interpreted by either the host PC or the robot's embedded computer. It should be noted that the Cricket ultrasonic transmitter and receivers are incorrectly labelled in the manual, 'US1' is the receiver. This must be taken account of when positioning the beacons.

$$c_{air} = 331.3\sqrt{1 + \frac{\theta}{273.15}} \tag{2.4}$$

The Cricket system can be configured in one of two ways. In one configuration three or more landmarks are configured as active beacons and a passive listener is mounted on the robot(s). This is shown graphically in Figure 2.22. Using active transmitters on the robots provides simultaneous measurement from each of the beacons, but means that the beacons must be connected to the host computer making the system rather cumbersome. Additionally, each robot has to take a turn to localise, leading to scaling issues for a large number of robots.

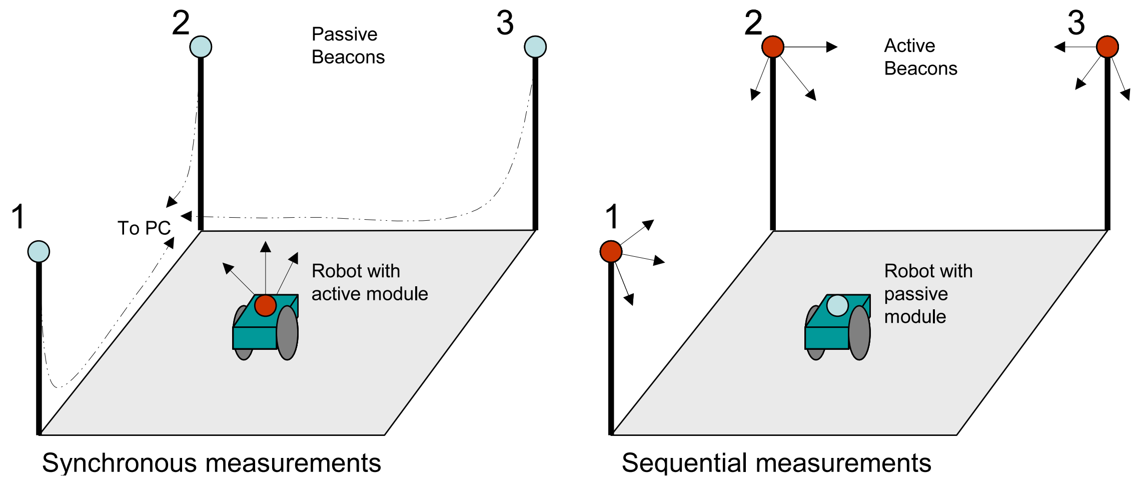


Figure 2.22: Two possible configurations of the Cricket Indoor Positioning System [74]

In the opposite configuration there are active modules on the robot(s) and the landmarks as passive listeners. In this configuration the landmarks are wireless modules, simplifying the system set up, however the measurements of distance between the robot and each beacon is now sequential. When the robot is moving, the distance to the first beacon will be significantly different by the time the measurement to the third beacon arrives. This degrades the result of any trilateration calculation if it is not properly taken into account.

As discussed in Chapter 3, superior performance was achieved with synchronised readings.

2.11.1.1 Passive Beacons, Simultaneous Measurements

In this configuration, the distance measurement is supplied by the listener beacons. USB to serial adaptors were used to create enough serial ports for all three beacons. The robot mounted active module is configured to periodically emit simultaneous RF and ultrasonic pulses. On reception of the pulses the listeners calculate the time distance and return this as a string to the PC. The GUI application captures these strings and parses them to obtain the active modules ID and the travel time. It then transmits the three distances to the robot over WiFi.

2.11.1.2 Active Beacons, Synchronous Measurements

In this configuration, the distance is supplied by the robot mounted listener. The active beacons are configured to pulse periodically, roughly falling into sequential order When the robot mounted module receives the signals it calculates the distances and transmits this over a serial link to the robot. Since the robot's serial hardware operates at UART voltage levels (0V - 5V) and not at the RS232 voltages available on the connector (-12V to 0V), a fly-wire was placed on the Cricket model to extract the signal before it was converted to RS232. The embedded computers UART's are required elsewhere, so the Cricket module was interfaced with the AVRmega128 micrcontroller then passed back to the embedded computer over I2C.

2.11.2 Visual Tracking

A positioning system with greater accuracy than the Cricket Indoor Positioning System was required to characterise robot tracking algorithms. Although finer accuracy was required, the system did not need to run in real time and did not require a large work area. It should be noted that this positioning system was a temporary solution for CUE and that it will be replaced by a commercial camera based system from Vicon [77] in the near future (not in time for this thesis). The intention was not to compete with the commercial system, but to provided a solution fit for purpose that would allow the work to proceed. That said, this work did give an interesting insight into the realisation of a camera based positioning system. An accuracy of 1-2mm was sufficient for benchmarking at this point.

The visual tracking system was created using a Logitech QuickCam Pro 9000 webcam and image processing techniques to extract the robot pose from each image. The work area was limited to the 0.8m × 1.2m visible in each image and all image processing was performed offline. The camera was configured to take one picture per second with a resolution of 960 × 720 pixels. A sample image is shown in Figure 2.23. A target was attached to the robot which could be accurately extracted from each image. The target is shown in Figure 2.24 and is clearly visible in Figure 2.23. The target contained seven black 10mm diameter discs on a white background. When photographed, the discs had a diameter of approximately nine pixels, which made them large enough for accurate extraction using image processing techniques. The triangular shape allowed the robot orientation to be determined from the orientation of the triangle. Strictly speaking, only three discs were required at the three corners of the triangle. The additional discs improved accuracy and robustness by providing additional data. It took 2-3 seconds to extract the robot pose from each image when the algorithm was run on a 2GHz, 2GByte RAM, Windows XP computer.

All image processing was performed in Matlab v7.1 [73] using the Image Processing Toolbox. The extraction of robot pose was a two step process. Firstly the target's 7 black discs were extracted from the image. The position of each of these discs was passed to a second algorithm that fitted a triangle to the points, then extracted the robot's pose from that of the triangle.
The seven discs were extracted using the following procedure.

1. The image was converted to grayscale and a threshold was used to convert it to a binary image, this is shown in Figure 2.25.

2. The next stage extracts the region boundaries which is relatively simple on a binary image. For convenience the Matlab algorithm ‘bwboundaries’ was used. The boundaries are shown in red in Figure 2.26.

3. The next stage is to classify each region and determine if it represents an element on the target. For each region, four parameters were calculated: ‘Extent’, ‘Length’, ‘Width’ and ‘Centroid’. The ‘Extent’ property returns the number of white pixels within the region’s bounding box of dimensions ‘Length’ × ‘Width’. The region is at the ‘Centroid’ location. Again, the Matlab Imaging Toolbox has a suitable algorithm, in this case ‘regionprops’.

4. The final stage was to examine the properties of each region to determine if it was a disc from the target. The discs had the following properties:

   - The length and width must be close to equal, since the bounding box of a disc should be a square.
   - The length and width must be of a specific size. The target is always a set distance from the camera, so the discs are a constant size.
   - The extent must be approximately 78.5% ($\frac{\pi}{4}$%), since a disc has 78.5% the surface of the bounding square with dimensions: diameter × diameter.

   Applying these conditions to each region reveals the discs shown in Figure 2.27. All discs on the target have been located, plus one spurious disc (on the lower handle of the wire strippers). If a maximum range of robot movement between frames was set, the algorithm’s speed was improved by only considering the portion of each image where the robot should be.

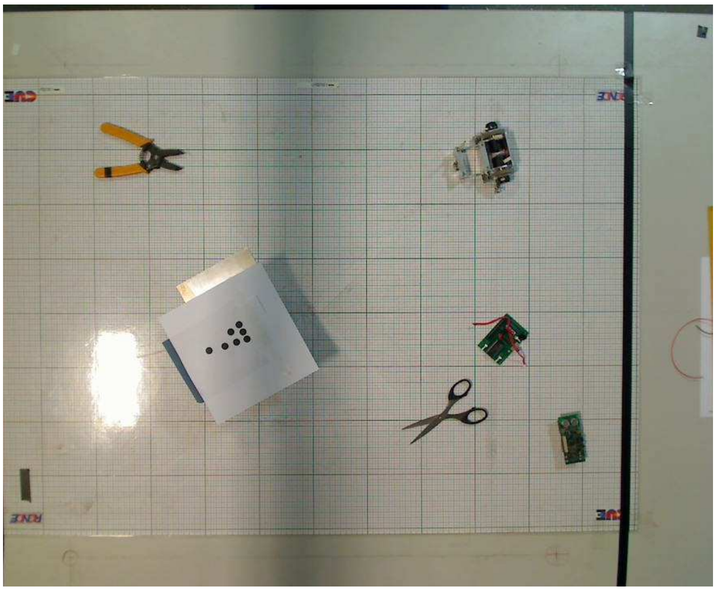


Figure 2.23: Sample image of the miniature robot with a target. Other objects are present to show the robustness of the approach

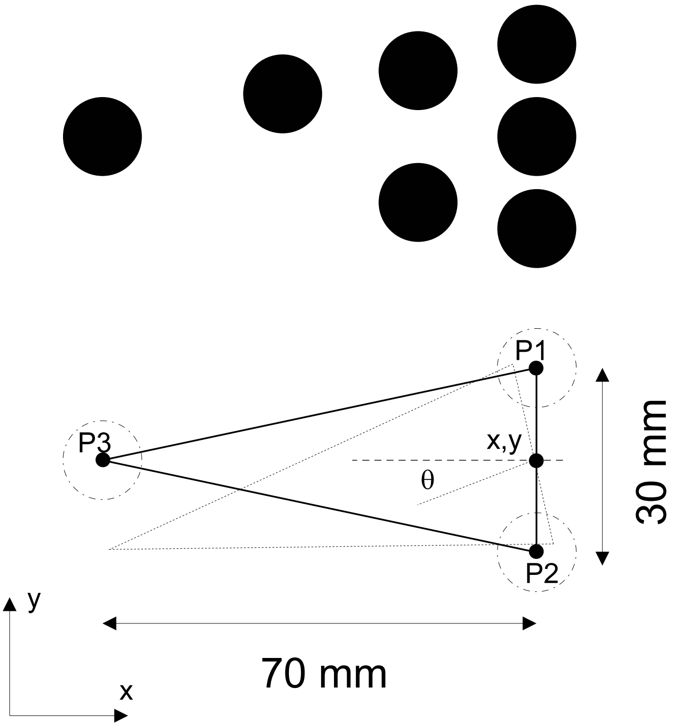


Figure 2.24: Target used to visually track robots position and orientation

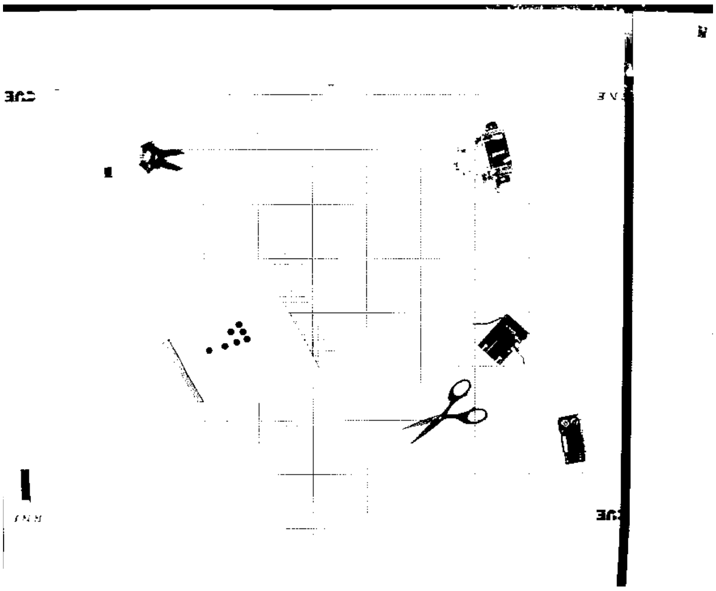


Figure 2.25: Sample image converted to a binary image

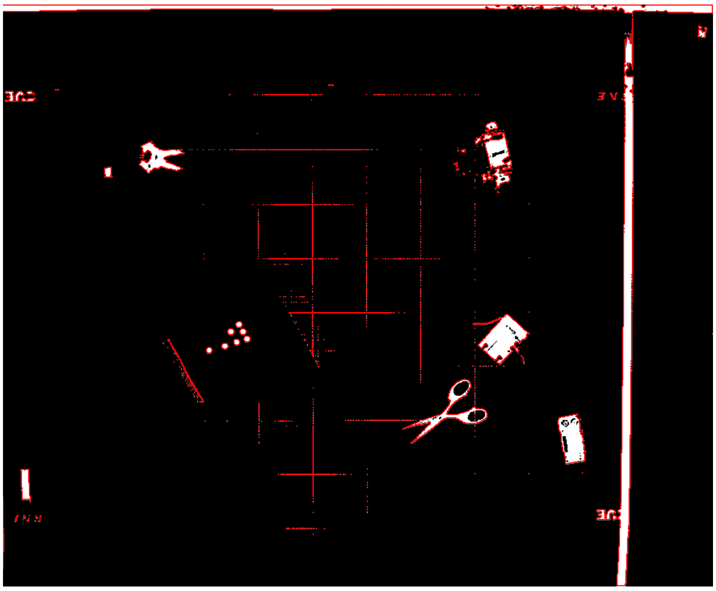


Figure 2.26: Sample image with regions boundaries marked in red

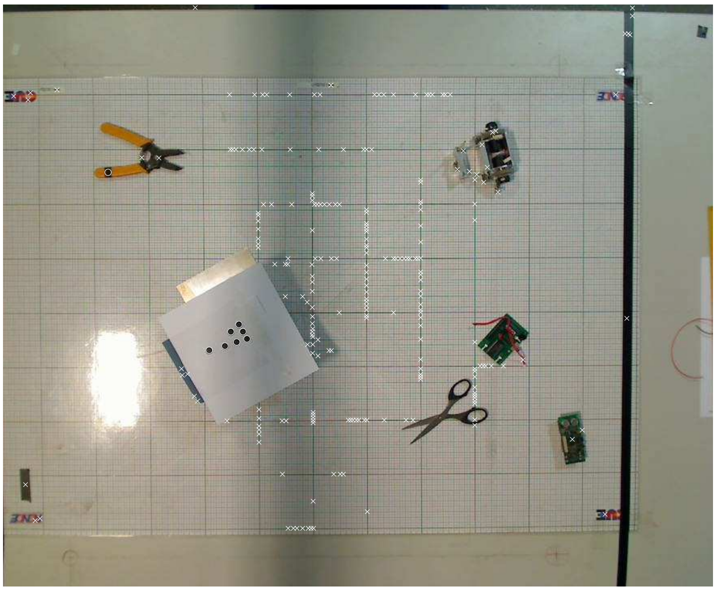


Figure 2.27: Sample image with regions identified, circles of the appropriate size are marked with an 'O', other features are marked with a '×'

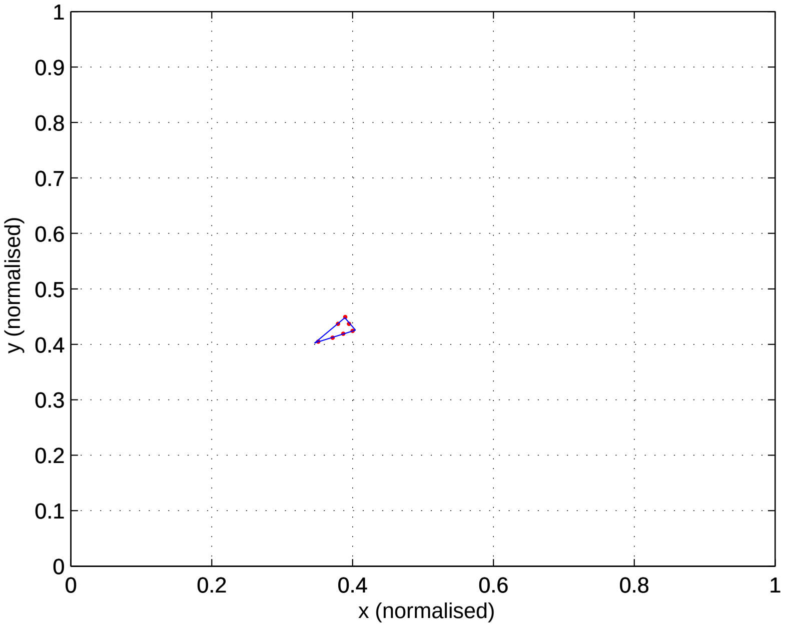


Figure 2.28: Robot pose identified

The centroid points of each disc was passed to the second algorithm that attempted to best fit a triangle to the points. The triangle is defined by three variables; the position in x and y, and the angle of rotation. With reference to Figure 2.24 the three corners of the triangle are located at:

$$P1_x = x - \frac{width}{2} sin(\theta) \qquad P1_y = y + \frac{width}{2} cos(\theta) \tag{2.5}$$

$$P2_x = x + \frac{width}{2} sin(\theta) \qquad P2_y = y - \frac{width}{2} cos(\theta) \tag{2.6}$$

$$P3_x = x - length \times cos(\theta) \qquad P3_y = y - length \times sin(\theta) \tag{2.7}$$

Where ‘X’, ‘Y’ is defined as the midpoint between P1 and P2. The length and width of the triangle are constant and matched to the size of the target in the image. The distance between each point and this triangle can be calculated by finding the shortest distance between the point and each of the three line segments that make up the triangle using simple geometry. The quality of the fit (Mean Squared Error, MSE) can be defined as the mean of the squares of the shortest distances between each point and the triangle. Standard optimisation techniques can be applied to solve for the triangle that has the best fit to the seven points. In this case Matlab’s standard unconditioned minimisation function, ‘fminunc’, was used. If more than 7 points have been extracted from the image, the optimisation runs several times, each time deleting the point with the highest MSE from the previous run until only 7 points remain. This approach works well at removing outliers. The quality of the final fit can be measured from the MSE, so that poor matches can be rejected. Occasionally the optimisation algorithm gets stuck in a local minimum and returns a poor fit, an example is shown in Figure 2.29. This can easily be identified by an unsatisfactorily high MSE. If the optimisation returns such a result, the starting condition of θ is increased by 45^o, while maintaining the X,Y coordinates from the last optimisation. The optimisation is then performed again. This is repeated until the MSE is below an acceptable threshold, in this case $3mm^2$.

If the point ‘X,Y’ is placed over the midpoint between the robot’s wheels and orientated to line up with the robot, the robot’s position and angle of rotation are equivalent to that of the triangle. If this alignment is not possible, a co-ordinate transform (Appendix B) can be used to transform from the world coordinates to robot coordinates, the offset removed, then transformed back to yield the robot's position in the global co-ordinate system. The co-ordinates are calculated as relative positions in the image and must be scaled back to millimetres. This can easily be achieved multiplying the relative position by the image dimensions in pixels to get a result in pixels, then multiplying by the number of millimeters per pixel to get the result in millimetres. The robot position calculated from Figure 2.23 is shown in Figure 2.28.

In order to validate the accuracy of this positioning system the robot was placed in 63 uniformly spaced positions on a 700mm x 900mm grid (100mm cells). At each position the an image was taken and the robot's pose calculated. The result is shown in Figure 2.30. Figures 2.31, 2.32 and 2.33 show the errors for all points in X, Y and θ respectively. The errors have a standard deviation of less than 0.85mm in X an Y and 0.54^o in θ. Increasing the resolution of the images would reduce this error, but increases the processing time. A portion of the error can be attributed to inaccurate manual positioning of the robot which is likely to have an error greater than $\pm 1mm$. If the algorithm was required to track several robots, the robots could be encoded with colour targets, these could easily be distinguished by considering the red, green and blue components of the raw image.

An example of tracking a single robot following a repeating square path is shown in Figure 2.34. The deviation from the ideal path is caused by a build-up of error in the positional estimate from the relative wheel encoder measurements, a technique for correcting for this is discussed in Chapter 3.

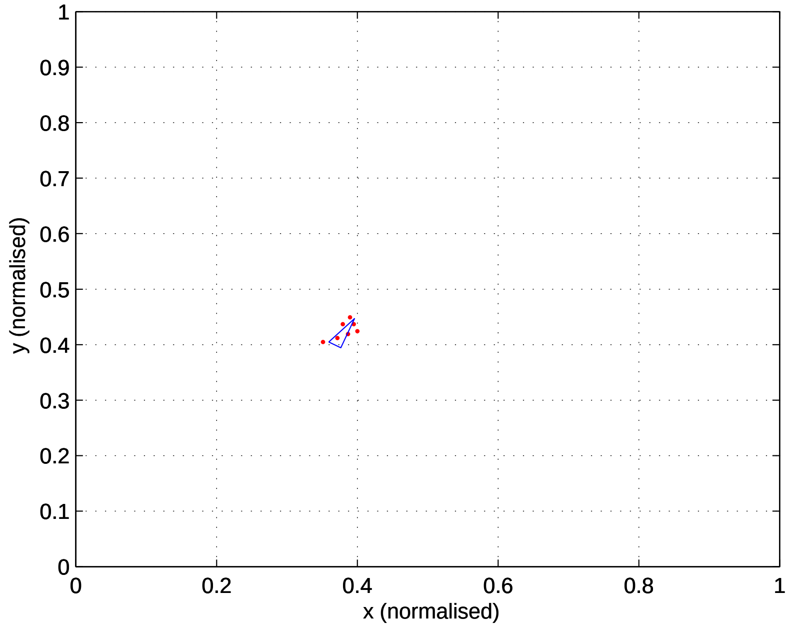


Figure 2.29: Robot pose identified incorrectly

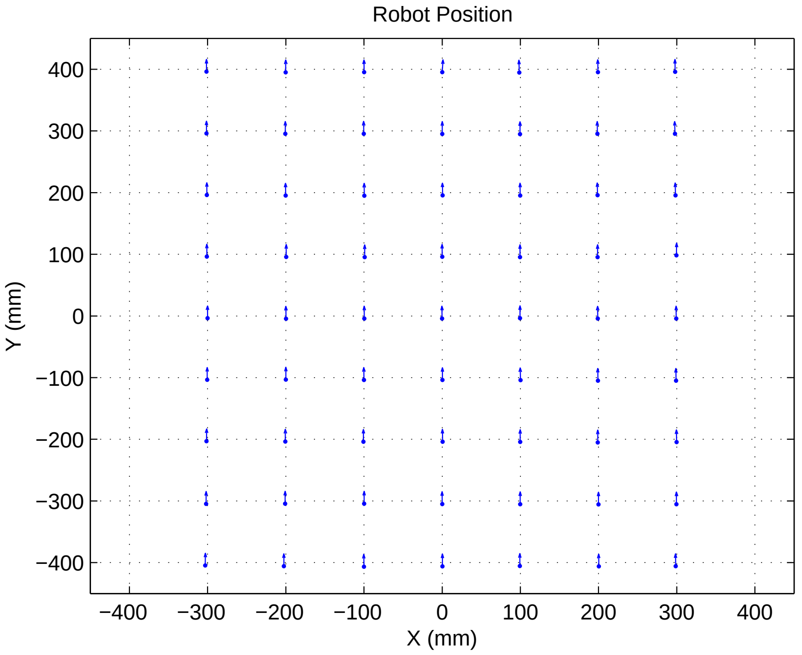


Figure 2.30: Visual positioning system validation. The robot positions should align with grid facing forward.

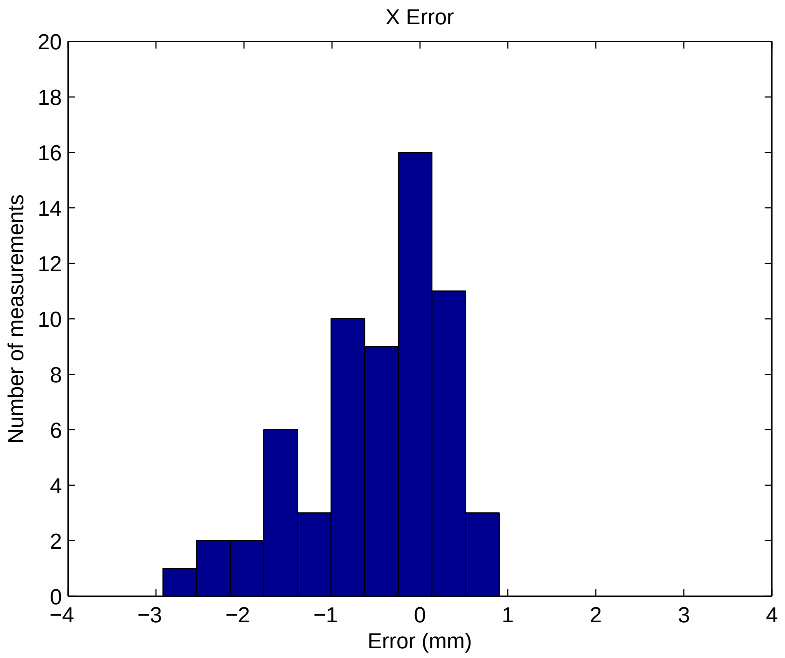


Figure 2.31: Visual positioning system validation. Histogram of error in X.

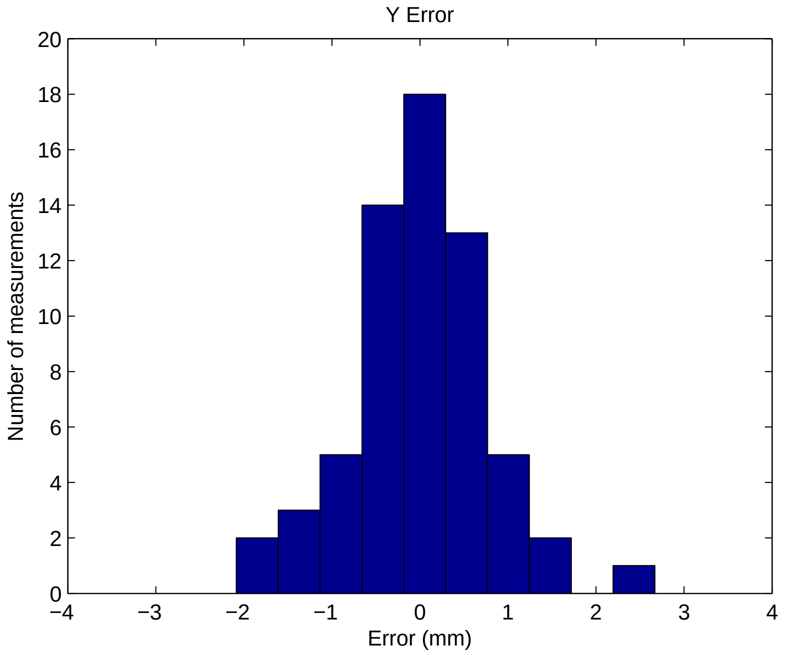


Figure 2.32: Visual positioning system validation. Histogram of error in Y.

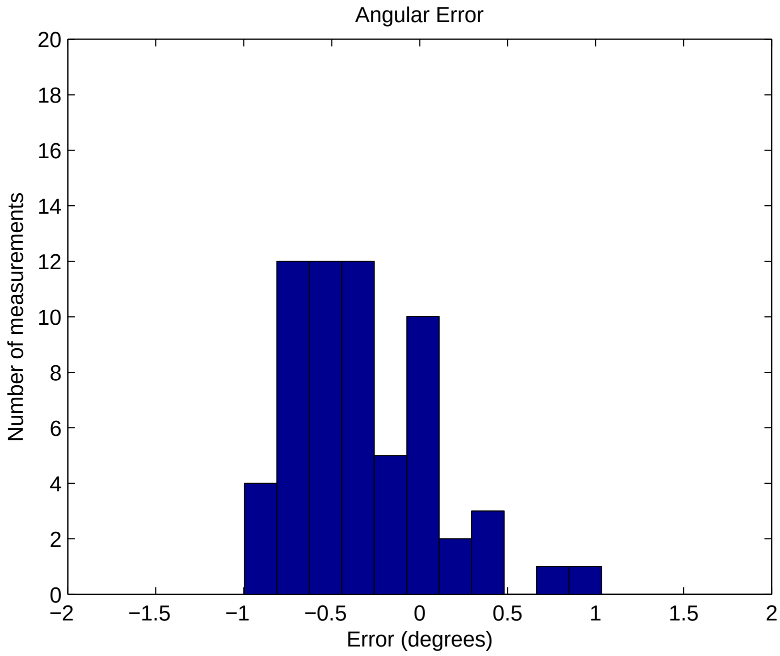


Figure 2.33: Visual positioning system validation. Histogram of error in θ.

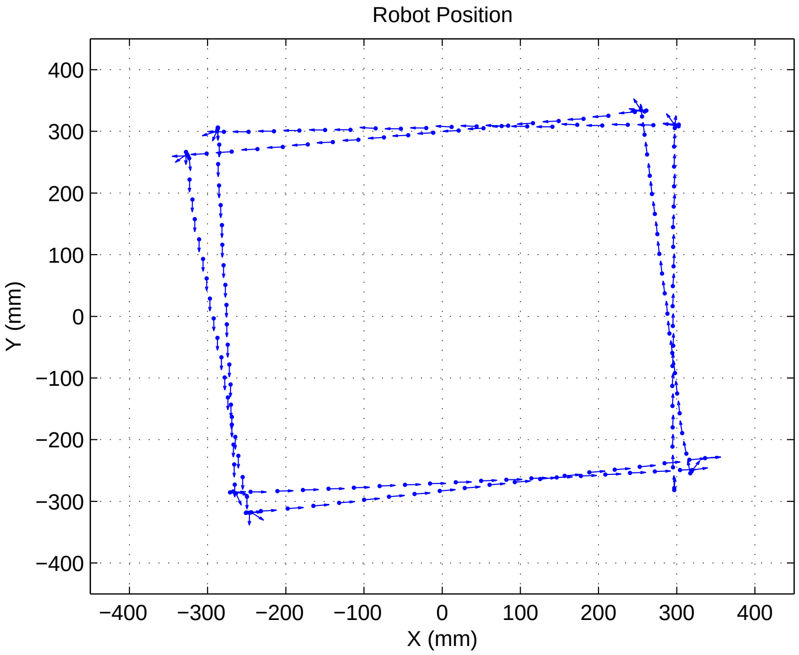


Figure 2.34: Example of robot tracking using visual positioning system

2.11.3 Visual Payloads

A visual payload was created by interfacing the C328 miniature camera [78] module with the embedded computer. The camera module has the following specifications:

- 4 Resolutions - 80 × 64, 160 × 128, 320 × 240 & 640 × 480 colour images.
- UART interface at 115.2 Kbps
- Low power - 60mA at 3.3v
- Integrated JPEG compression
- Compact size (20 × 28 mm)

A C++ driver was written that used one of the embedded computer's UARTs to communicate with the camera module. This task was initially given to an intern student, then formed part of a 5th year group project [79], before being refined as part of this work. Capturing images(640 × 480) takes approximately one second, in order to prevent this from stalling the main program this process runs in a separate thread. The camera module uses an auto baud rate algorithm that requires the synchronisation of the host and module. During initialisation a 12 byte sync command is repeatedly sent to the module until it replies with the 12 byte acknowledge command. This can take up to 60 iterations, but is usually complete in 25. The camera is then configured, setting the image resolution and packet size. On requesting a picture, the camera takes and compresses the picture, then returns the image size. The image is returned in multiple packets of length 'packet size'. The length can be determined by dividing the image size by the packet size minus six bytes for a header and footer in each packet. This is summarised in Figure 2.35.

Figure 2.10 shows the camera module mounted on the robot. Figure 2.36 shows an example picture. Section 2.10.3 discuss the presentation of these NDE results.

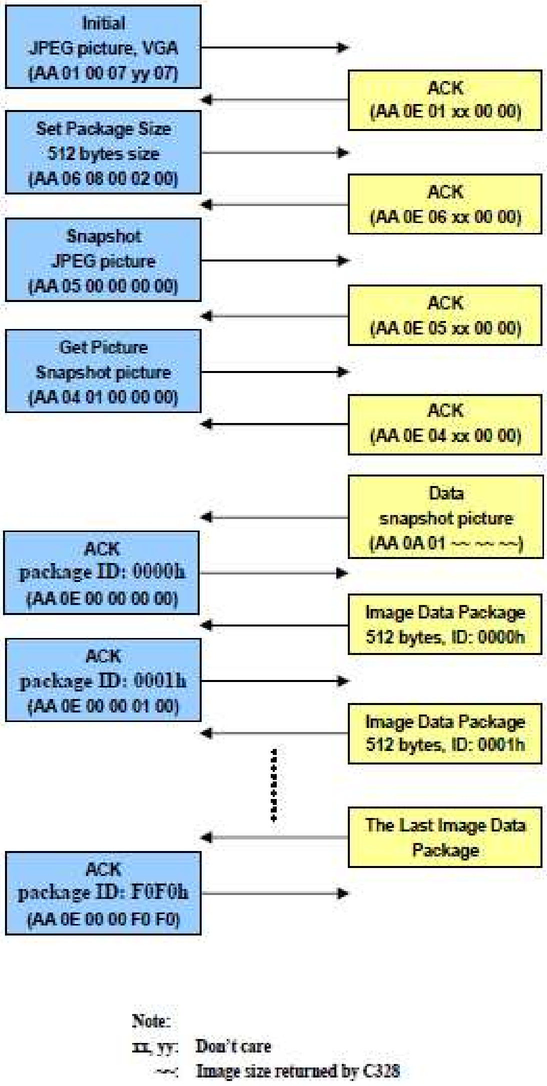


Figure 2.35: Flow chart showing interface with C328 camera module, taken from the data sheet [78]

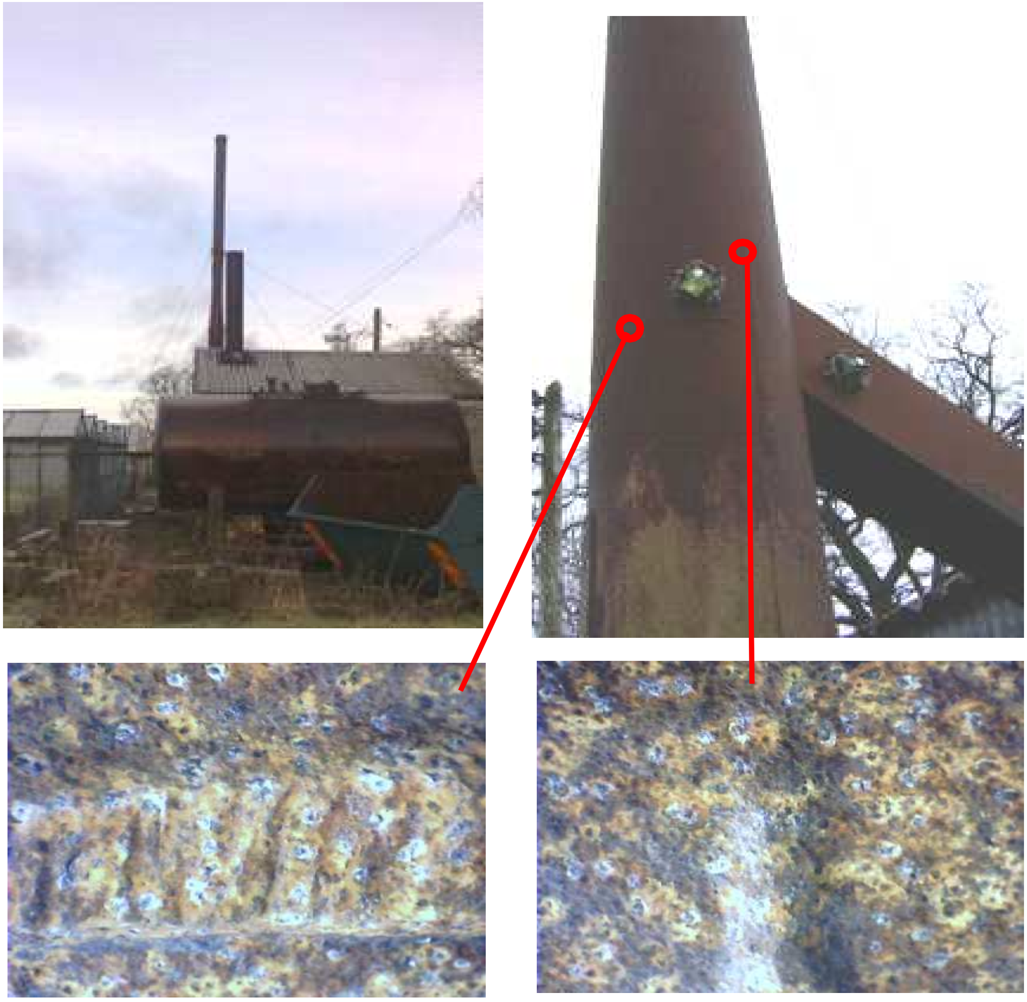


Figure 2.36: Example visual inspection results. The sample was a 25m steel chimney.

2.11.4 Remote Control

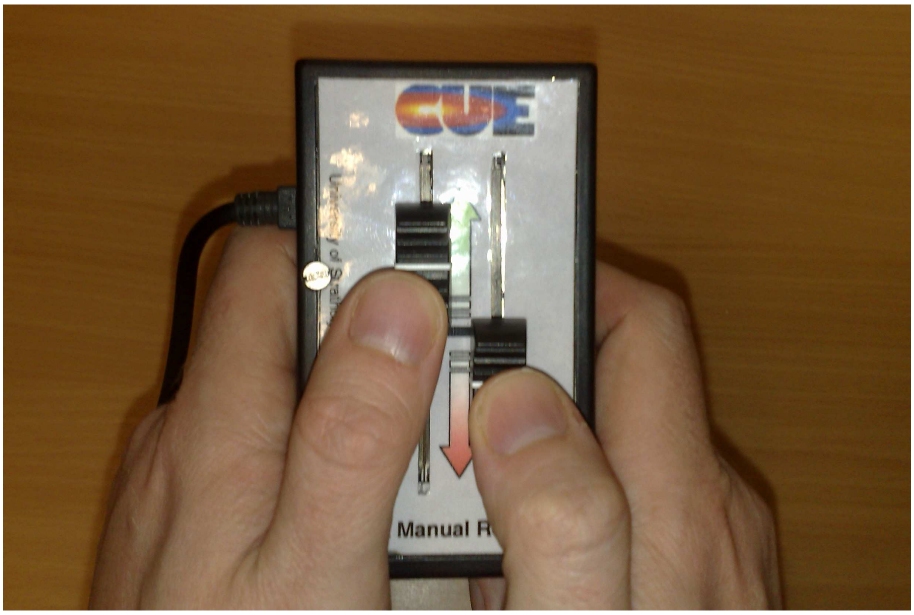


Figure 2.37: Photograph of USB remote control

Manually controlling a differential drive robot with buttons or software sliders is very difficult. The point and click control interface shown in Figure 2.20 is not always practical, particularly on 3D test specimens that have not been imported into the virtual world. To allow rapid testing of the robotic hardware on new specimens a hardware remote control was developed, this is shown in Figure 2.37. The remote control plugs into the host computers's USB port and communicates with the robots via the computer's WiFi. It consists of two vertical potentiometers, that are read by an embedded microcontroller and transmitted to the GUI application through USB. Each slider controls the wheel velocity directly, with the velocity proportional to the slider distance from the centre position.

2.12 Summary

This chapter has detailed the work creating a development platform for robotic NDE. The platform has the following features:

- Miniature differential drive robotic vehicles with magnetic wheels for traction. Each robot contains a 400 MHz Linux computer with WiFi for extensive embedded processing. The robots have a run-time of approximately 4 hours and can carry several NDE payloads.
- Embedded firmware to allow low level control and monitoring of the robots. This includes the facility to store robot settings on board allowing each robot to be configured independently.
- A PC host application that presents virtual representations of the robots in a virtual 3D world allowing intuitive control and monitoring of the fleet.
- Integration of a robot mounted camera module to form a visual inspection payload.
- Integration of NDE results with virtual representation of structure under test to facilitate in-context presentation of NDE results.
- Integration of Cricket global positioning system for the robot positioning research covered in Chapter 3.
- Integration of an accurate, but offline, visual positioning system for the characterisation of positioning algorithms. This is also used in Chapter 3.
- Creation of a handheld remote control for intuitive control of individual robots in unknown environments.

Chapter 3

Robot Positioning and Control

3.1 Introduction

Robotic positioning is a fundamental component for the successful realisation of the NDE scanner concept. Positioning with a good degree of accuracy is needed to generate robot paths, to avoid collisions between robots and to keep the robots in the designated inspection zone. In order to effectively interpret or combine a series of NDE measurements, the position where each measurement is taken is fundamental. Additionally, thorough sample coverage necessitates good positioning. Chapters 6 and Chapter 7 implement two different NDE scanning strategies, in both cases the quality of results is dependent on robot positional accuracy.

Consider a single RSA instructed to perform a scan of a large plate. If the initial location is known, the RSA can estimate its movement by recording the motion of the two drive wheels, either by monitoring the motor drive signal or more accurately using wheel encoders. This is known as 'dead reckoning' and is the most basic form of localisation. The measurements are imperfect due to wheel slippage and other subtle noise sources [80]. These errors accumulate over time increasing the uncertainty in the position estimate. Although accurate over short periods, dead reckoning is not adequate for large scans. The problem of accumulating errors can be obviated by the use of absolute positional sensors such as a Global Positioning System (GPS). (Note that in this work GPS is used to represent any global positioning system and not necessarily the satellite GPS system created by the United States.) However, absolute positioning systems also contain uncertainty in their measurements and typically do not have the fidelity required for adequate tracking. Absolute positioning sensors may only provide partial information, such as position, but not orientation.

It is possible to use a Kalman Filter [12] to optimally combine the good short term accuracy of relative measurements with less accurate absolute measurements. This approach has been widely utilised in the robotics community [11, 13, 14, 15, 81].

This Chapter focuses on the derivation, simulation and implementation of a Kalman Filter into the RSA platform. The Chapter begins by defining the robot localisation problem in Section 3.2. Sections 3.3 and 3.4 discuss the merits of different sensor modalities. The theory behind the Kalman Filter is presented in Section 3.5. Simulation of the RSA is presented in Section 3.6 and finally implementation and experimental results are demonstrated in Section 3.7.

The contribution of this chapter was the successful implementation of an Extended Kalman Filter into the RSA platform. The work did not concern itself with the derivation of novel positioning algorithms, but instead focused on implementation and benchmarking. This was an enabling step, that achieved two goals. Firstly, it provided reasonable RSA positional accuracy (peak error of 10 mm) that allowed the realisation of other parts of this system. Secondly, it provided a benchmark against which other RSA positioning algorithms will be compared.

3.2 Robot Localisation

Robot localisation is the determination of the robots position and orientation (generally refereed to as the robots pose) relative to its environment. It is one of the most fundamental problems in mobile robotics. It can be broken down into three successively difficult problems [81].

1. Position tracking given a known initial location
2. 'Wake-up' problem. As above, but without a known initial location.
3. 'Kidnapped robot' problem where the localisation algorithm must be able to recover from the robot being secretly moved to an unknown location.

In the current application it can be assumed that the initial position of the RSA is known and that it will not be 'kidnapped'.

3.3 Relative Position Measurement

Relative positional sensors provide a relative update to the robot position, e.g. the robot has travelled 10mm since the last measurement. By integrating these measurements, it is possible to track the robot's location. Relative positional sensors can be divided into two categories, inertial and odometric [82].

3.3.1 Inertial Sensors

Inertial sensors measure the rate of rotation (gyroscopes) or acceleration (accelerometers). Recent advances in MEMS technology have lead to compact and affordable devices. For example the ADIS16364 from Analog Devices [83] combines a $\pm 75^o/sec$ three axis gyroscope with a $\pm 5g$ three axis accelerometer in a package measuring only $23 \times 23 \times 23$ mm. For single axis gyroscopes, such as the ADXRS613, devices can be as small as $7 \times 7 \times 3$ mm. Inertia measurements need to be integrated, so that small errors increase without bound [82], making them unsuitable for accurate positioning over an extended period. As discussed in Chapter 2 the RSA has an ADXRS613 gyroscope, from Analog Devices [83], that detects rotation in the horizontal plane.

3.3.2 Odometry Sensors

Odometric sensors, such as wheel encoders, provide a relative measurement of wheel rotation. By combining these measurements with a static model of the robot, it is possible to track the robot's pose.

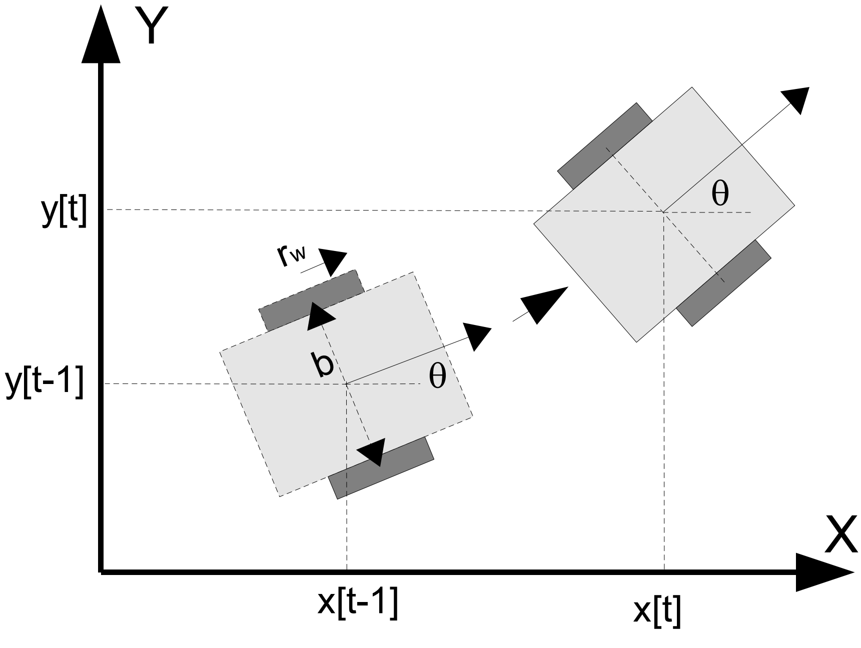


Figure 3.1: Dead reckoning calculation based on two odometric sensors [1]

Figure 3.1 shows a snapshot of the robot's pose at two successive time instances, 't-1' and 't'. At each discrete time interval the distance travelled by the two wheels can be determined by recording the number of pulses received from the optical encoders. The wheels have a diameter, 'D' and are separated by a distance 'b'. The pose, 'θ' is defined as the angle from the X axis, as shown in Figure 3.1. If the time interval 'Δt' is small, the robot's pose can be estimated as follows:

$$x[t] = x[t-1] + \frac{(w_r C_r + w_l C_l) cos(\theta[t-1])}{2} \tag{3.1}$$

$$y[t] = y[t-1] + \frac{(w_r C_r + w_l C_l) sin(\theta[t-1])}{2} \tag{3.2}$$

$$\theta[t] = \theta[t-1] + \frac{w_r C_r - w_l C_l}{b} \tag{3.3}$$

Where ω_l and ω_r are the number of encoder pulses recorded since the last update for the left and right wheels respectively. The wheel conversion factors (C_l and C_r) are calculated from the wheel diameter, D, and the encoder resolution, R, using Equation 3.4. The four on the denominator accounts for quadrature encoding (See Section 2.9.1, Chapter 2).

$$C_l = \frac{\pi D}{4R} \tag{3.4}$$

Odometric sensing has excellent short term accuracy, is inexpensive and has relatively high sample rates [82]. However, the sensors cannot account for wheel slippage, errors in the robot's mechanics (D or b), or uneven surfaces. The measurements are subject to an integral error that grows without bound.

Dead-reckoning errors can be categorised as systematic errors and non-systematic errors [80]. Systematic errors result from an inaccuracies in the description of the robot mechanics, such as inaccurate wheel diameters or wheel base. Non-systematic errors include the more unpredictable interaction between the robot wheels and the surface leading to slippage, or motion over bumps or cracks. Using a calibration procedure called UMBmark, developed by Borenstein and Feng [84], the systematic errors can be minimised. UMBmark is discussed further in Section 3.8.1.

As discussed in Chapter 2, the RSA has an E4P encoder on each drive shaft. The E4P from US Digital [85] is the smallest wheel encoder currently available. It provides 300 count quadrature encoding in a 23mm x 23mm x 12mm package. This package makes the encoders rather large for use in miniature robots, particularly since they need to be attached to the drive shaft, limiting the scope to optimise the robot's layout.

3.3.3 Dead-Reckoning Example

Consider a two wheeled, differential drive mobile robot, as discussed previously. The wheel diameter is 40mm and the distance between the wheels is 70mm. The robot is fitted with an optical encoder on each wheel that can measure the wheel rotational distance. It is assumed that the error in the encoder reading has a Gaussian distribution with a variance of 0.1 rad/s. The robot is programmed to follow a raster path shown in Figure 3.2. The wheel speed is 1.2 rad/s. The robot turns on the spot by driving the wheels in opposite directions at the same speed (±1.2 rad/s). The measured path is shown in Figure 3.2 and the XY mean square error is shown in Figure 3.3. The increasing MSE trend is a result of the integrating error.

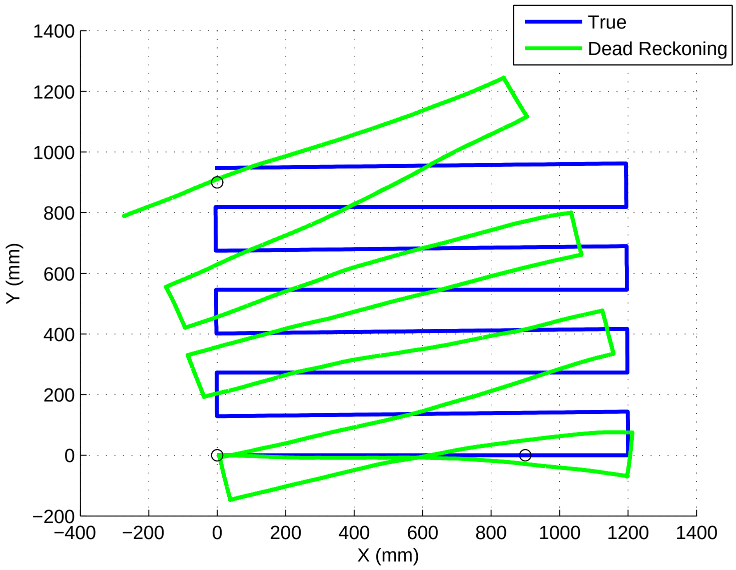


Figure 3.2: Positional inaccuracies due to error in wheel encoder readings, shown as absolute position.

This error can be reduced by improving the accuracy and number of sensors, but the accumulating uncertainty rules out using only relative sensors in practical systems.

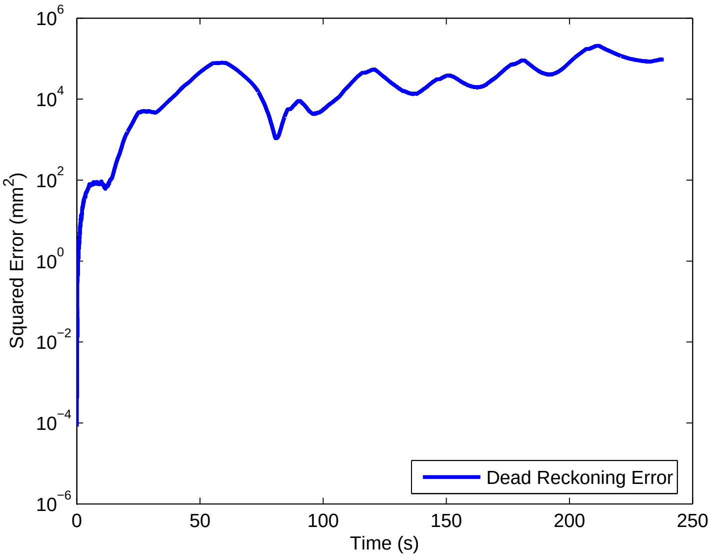


Figure 3.3: Positional inaccuracies due to error in wheel encoder readings, shown as squared error.

For this reason, relative sensors are often complemented with a Global Positioning System.

3.4 Global Positioning System

Most GPS systems use one of the following techniques: Triangulation (lateration or angulation), scene analysis (such as landmark recognition) or proximity detection (i.e. close to a known location) [86]. These systems use a wide range of sensing technologies including ultrasonic and lasers for triangulation distance measurements, image processing techniques for scene analysis and simple contact sensors for proximity detection. A range of factors contribute to the applicability of these systems, including accuracy, update rate, scale, cost and size, all of which are discussed in excellent reviews by Hightower and Borriello [86] and Borenstein et al.[82].

The Cricket Indoor Positioning System [74, 75] was used to provide a global reference. An overview of Cricket and its hardware implementation are discussed in Section 2.11.1, Chapter 2.

3.4.1 Triliteration Equations

Using three fixed Cricket beacons, it is possible to calculate the position of a robot mounted Cricket module. The beacons are positioned on three corners of an imaginary square with length 'r' on a horizontal plane, i.e. $b_1 = [0, 0, 0]$, $b_2 = [r, 0, 0]$ and $b_3 = [0, r, 0]$ in $[x, y, z]$ dimensions. The position of the robot mounted module can be calculated as:

$$x = \frac{1}{2r}(d_1^2 - d_2^2 + r^2) \tag{3.5}$$

$$y = \frac{1}{2r}(d_1^2 - d_3^2 + r^2) \tag{3.6}$$

$$z = \pm\sqrt{(d_1^2 - x^2 - y^2)} \tag{3.7}$$

Where d_1, d_2 and d_3 are the distances between the robot module and beacons 1, 2 and 3 respectively. Since the beacons are positioned above the robot work area, the negative result for 'z' can be used. The performance of the Cricket Indoor Positioning System is characterised in Section 3.7.1.

3.5 Kalman Filtering Theory

A Bayes filter is the most general algorithm for calculating beliefs. It represents belief distribution using both control and measurement data. The Bayes filter is a recursive algorithm that can be represented as [81]:

```
for all $x_t$ do:
        $\bar{bel}(x_t) = \int p(x_t|u_t, x_{t-1})bel(x_{t-1})dx_{t-1}$
        $bel(x_t) = \eta p(z_t|x_t)\bar{bel}(x_t)$
end for loop
```

The belief $bel(x_t)$ is calculated from the previous belief $bel(x_{t-1})$, the control u_t and the measurement z_t. The Bayes filter is a two stage process. Firstly, the filter propagates the previous belief forward a time step based on the control input, this is known as the *prediction.* This prediction is the integral of the probability density functions for the prior belief and the probability, $p(x_t|u_t, x_{t-1})$, that the control induces the transformation from x_{t-1} to x_t. The second stage multiplies this belief $\bar{bel}(x_t)$ by the probability that the measurement, z_t, has been observed, $p(z_t|x_t)$. This is known as the *measurement update* or *correction.* The result is not technically a probability, since it does not sum to 1. It is therefore normalised using the normalisation constant η.

The Bayes filter relies on the Markov assumption, i.e. that past and future states are independent if the current state is known. The ramifications of the Markov assumption are discussed in [81] and will not be repeated here. The Kalman Filter is the best studied technique of practically implementing a Bayes filter [81].

The Kalman filter is an optimal implementation of a Bayes filter for linear systems. Beliefs are represented by their mean and covariance. The state transition probability must be a linear function. The Kalman filter minimises the mean square error (MSE) in the posterior belief's covariance. It is useful to begin with a basic example before consolidating with a mathematical description. The following example, which has been adapted from [87], considers an object moving at constant speed ẋ. Assuming the sampling frequency is a constant, the robot will travel $u_t = \dot{x}dt$, hence:

$$x_t = x_{t-1} + u_t + \omega_t \tag{3.8}$$

The new location x_t is calculated recursively, with the inclusion of a noise term ω_t to account for inaccuracies in the model. The noise is assumed to be random, zero-mean with a Gaussian distribution. We have an initial estimate of the object's position as $\hat{x}_0$. The uncertainty in the initial estimate can be specified as a variance ${\sigma^2}_{t=1}$.

After a single time step, the object's position can be estimated as:

$$\hat{x}_{t=1} = \hat{x}_{t=0} + u_t + 0 \tag{3.9}$$

It is not possible to know the value of the system model noise, so the mean (0) is used. Calculation 3.9 is known as the Kalman Filter *prediction* where the state of the object is predicted recursively from the input. The propagation of uncertainty is described by Equation 3.10.

$$\sigma^2_{t=1} = {\sigma^2}_{t=0} + {\sigma^2}_{\omega} \tag{3.10}$$

σ^2_t is defined as the variance of the uncertainty in $\hat{x}_t$. As shown in Equation 3.10, relying solely on the model would result in an uncertainty that grows without bound. This can be also seen in the odometry example in Section 3.3.3.

Consider now an absolute measurement z_t, which is somewhat corrupted by noise υ_t, i.e.

$$z_t = x_t + \upsilon_t \tag{3.11}$$

Again, the noise is assumed to be random, zero-mean, with a Gaussian distribution. The measurement noise is characterised by the variance ${\sigma^2}_{\upsilon}$. The key to the Kalman Filter is using this measurement to *correct* the prediction made by Calculation 3.9. This is achieved using a weighted average based on the uncertainty in the system model and uncertainty in the measurement.

$$\begin{aligned} {\hat{x}_t}^+ &= \tfrac{{\sigma_\upsilon}^2}{{\sigma_t}^2+{\sigma_\upsilon}^2}\hat{x}_t + \tfrac{{\sigma_t}^2}{{\sigma_t}^2+{\sigma_\upsilon}^2}z_t & (3.12) \\ &= \hat{x}_t + \tfrac{{\sigma_t}^2}{{\sigma_t}^2+{\sigma_\upsilon}^2}(z_t - \hat{x}_t) & (3.13) \end{aligned}$$

This equation can be simplified by introducing the Kalman Gain, which in this case can be written as:

$$K = \frac{{\sigma_t}^2}{{\sigma_t}^2 + {\sigma_v}^2} \tag{3.14}$$

Consolidating this example, we arrive at the Kalman Filter equations. Equation 3.15 defines the prediction stage and Equation 3.17 defines the measurement correction. The Kalman Gain is recalculated at each iteration to minimise the uncertainty in ${x_t}^+$.

$$\hat{x}_t = x_{t-1} + u_t + \omega_t \tag{3.15}$$

$$z_t = x_t + \upsilon_t \tag{3.16}$$

$${x_t}^+ = K(z_1 - \hat{x}_1) \tag{3.17}$$

The uncertainty in the prediction ${x_t}^+$ can be written as:

$$\frac{1}{{\sigma_t}^{2,+}} = \frac{1}{{\sigma_t}^2} + \frac{1}{{\sigma_v}^2} \tag{3.18}$$

and simplified to:

$${\sigma_t}^{2,+} = (1 - K){\sigma_t}^2 \tag{3.19}$$

Equation 3.19 shows that the measurement always acts to reduce the uncertainty in ${x_t}^+$.

3.5.1 The Extended Kalman Filter

Propagation of a Gaussian through a linear transformation produces another Gaussian. The Kalman filter requires an initial Gaussian belief, linear state transition and Gaussian measurement probabilities ($p(x_t|u_t, x_{t-1})$ and $p(z_t|x_t)$). In which case, the transition of the $\hat{bel}(x_{t-1})$ Gaussian produces a new Gaussian $\hat{bel}(x_t)$. In order to use the Kalman Filtering approach to predict the state of non-linear stochastic equations (such as a differential drive robot) the Extended Kalman Filter (EKF) must be used. The EKF linearises around the current mean and covariance at each iteration and then essentially performs like a regular Kalman Filter. The severity of this approximation is system dependent, but since the Bayes filter does not have a closed-form solution for non-linear system or measurement models, an approximation is necessary [81]. The EKF uses a first order Taylor expansion of the non-linear function, which approximates the non-linear function to a linear function passing through the same operating point, with a gradient equal to the non-linear function at that point. For reasons of brevity and the similarity in the mathematics of Kalman Filter and EKF, only the EKF will be discussed. The reader should refer to the excellent introduction by Welch and Bishop [88] or a book by Thrun et al.[81] for more information about the Kalman Filter and EKF.

Consider a process that is governed by the non-linear stochastic difference equation, with $x \in \Re^n$ (n variables)

$$x_t = f(x_{t-1}, u_{t-1}, w_{t-1}) \tag{3.20}$$

x_t represents the next state with respect to the previous state x_{t-1}, an input u_{t-1} and the random variable w_{t-1} which represents process noise.

The system can be measured by:

$$z_t = h(x_t, v_t) \tag{3.21}$$

The random variable v_t represent measurement noise. Like the Bayes filter, the EKF is divided into two sets of equations; the prediction and the measurement update. The time prediction equations are defined as:

$$\hat{x}_t^- = f(x_{t-1}, u_{t-1}, 0) \tag{3.22}$$

$$P^-{}_t = A_t P_{t-1} {A_t}^T + W_t Q_{t-1} {W_t}^T \tag{3.23}$$

Equation 3.22 calculates the a priori state estimate based on the inputs. Equation 3.23 predicts the error, $P^-{}_t$, in the a priori estimate error covariance $conv(x_t - \hat{x}_t^-)$. A_t and W_t are the process Jacobians at time step t. Q_t is the process noise covariance at time step t. The Jacobians linearise the system around the operating point t.

The measurement update equations are defined as:

$$\hat{x}_t = \hat{x}_t^- + K_t(z_t - h(\hat{x}_t^-, 0)) \tag{3.24}$$

$$K_t = P^-{}_t {H_t}^T (H_t P^-{}_t {H_t}^T + V_t R_t {V_t}^T)^{-1} \tag{3.25}$$

$$P_t = (I - K_t H_t) P^-{}_t \tag{3.26}$$

H_t and V_t are the measurement Jacobians at time step t and R_t is the measurement noise error covariance. Equation 3.24 corrects the a priori state estimate using the measurement z_t. The influence of the measurement is dependent on the Kalman gain which is calculated in Equation 3.25. The Kalman gain is calculated to minimise the posteriori error covariance; a derivation can be found in [81]. The Jacobian H_t serves to correctly propagate only the relevant component of the measurement information, i.e. only the part that affects the individual states [88]. The error covariance is propagated forward using Equation 3.26. The next section shows an example of applying the filter to a differential drive robot.

3.5.2 Application to Differential Drive Robot

The robot is described by four state variables. The first three states are the robot's Cartesian position in 3D, the last is the angle in the X-Y plane.

$$x_t = \begin{pmatrix} x \\ y \\ z \\ \theta \end{pmatrix} \tag{3.27}$$

The EKF is usually simplified by replacing the inputs u_t with the wheel encoder measurements, obviating the need for a complex model of the robot and its surrounds.

The system model is taken from the robot model in Section 3.3.2. It is presented in matrix form:

$$\begin{pmatrix} x_t \\ y_t \\ z_t \\ \theta_k \end{pmatrix} = \begin{pmatrix} x_{t-1} + \frac{1}{2}((\omega r + \omega r_{noise})C_r + (\omega l + \omega l_{noise})C_l)cos(\theta_{k-1}) \\ y_{t-1} + \frac{1}{2}((\omega r + \omega r_{noise})C_r + (\omega l + \omega l_{noise})C_l)sin(\theta_{k-1}) \\ z_{t-1} \\ \theta_{k-1} + \frac{1}{b}((\omega r + \omega r_{noise})C_r - (\omega l + \omega l_{noise})C_l) \end{pmatrix} \tag{3.28}$$

The two inputs, ω_l and ω_r, are the angles turned in the last time step, by the left and right wheels respectively. Each wheel has a respective noise component. Notice that the robot model cannot provide any information about the 'z' coordinate.

The Jacobians A_k and W_k are calculated as follows:

$$A_t = \frac{\partial f}{\partial x}|_{x=\hat{x}_{t-1},u=u_{t-1},w=0} = \begin{pmatrix} 1 & 0 & 0 & \frac{-1}{2}(C_r\omega r_{t-1} + C_l\omega l_{t-1})sin(\theta_{t-1}) \\ 0 & 1 & 0 & \frac{1}{2}(C_r\omega r_{t-1} + C_l\omega l_{t-1})cos(\theta_{t-1}) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{3.29}$$

$$W_t = \frac{\partial f}{\partial w}|_{x=\hat{x}_{t-1},u=u_{t-1},w=w_{t-1}} = \begin{pmatrix} \frac{C_l}{2}cos(\theta_{t-1}) & \frac{Cr}{2}sin(\theta_{t-1}) \\ \frac{C_r}{2}cos(\theta_{t-1}) & \frac{Cl}{2}sin(\theta_{t-1}) \\ 0 & 0 \\ \frac{C_l}{b} & \frac{C_r}{b} \end{pmatrix} \tag{3.30}$$

The process noise covariance matrix depends on the accuracy of the wheel encoders. This is defined as:

$$Q_t = \begin{pmatrix} {\sigma_{\omega L}}^2 & 0 \\ 0 & {\sigma_{\omega R}}^2 \end{pmatrix} \tag{3.31}$$

The Cricket measurement is now used to correct the prediction. In one form, the Kalman Filter measurement model is based on absolute distance measurements from a global positioning system with N beacons positioned in Cartesian space at $b_n(x, y, z)$ where $b_n \in \{b_1, ..., b_N\}$. The advantage of this approach is that the number and location of the beacons is not constrained.

$$h(x_t, v_t) = \begin{pmatrix} \sqrt{(x - b_1[x])^2 + (y - b_1[y]^2) + (z - b_1[z])^2} + v_1 \\ ... \\ \sqrt{(x - b_N[x])^2 + (y - b_N[y]^2) + (z - b_N[z])^2} + v_N \end{pmatrix} \tag{3.32}$$

The Jacobian matrix H_t can be calculated as follows:

$$H_t = \frac{\partial h}{\partial x}|_{x=\hat{x}_t, v=0} = \begin{pmatrix} \frac{\hat{x}^-[x]-b_1[x]}{L_1} & \frac{\hat{x}^-[y]-b_1[y]}{L_1} & \frac{\hat{x}^-[z]-b_1[z]}{L_1} & 0 \\ ... & ... & ... & \\ \frac{\hat{x}^-[x]-b_N[x]}{L_N} & \frac{\hat{x}^-[y]-b_N[y]}{L_N} & \frac{\hat{x}^-[z]-b_N[z]}{L_N} & 0 \end{pmatrix} \tag{3.33}$$

where

$$L_n = \sqrt{(\hat{x}^-[x] - b_n[x])^2 + (\hat{x}^-[y] - b_n[y])^2 + (\hat{x}^-[z] - b_n[z])^2} \tag{3.34}$$

The Jacobian V_k is the identity matrix, size N:

$$V_t = \frac{\partial h}{\partial v}|_{x=\hat{x}_t, v=v_t} = I(N) \tag{3.35}$$

Finally the measurement noise is assumed to be constant, so:

$$R = \begin{pmatrix} \sigma_{b1}^2 & ... & 0 \\ ... & \sigma_{bn}^2 & ... \\ 0 & ... & \sigma_{bN}^2 \end{pmatrix} \tag{3.36}$$

An alternative, simpler form of the EKF measurement model provides the filter with GPS positions calculated from trilateration, rather than the distance between the robot and each beacon. Using three beacons, positioned in specific locations, it is possible to analytically solve for X,Y and Z using the trilateration equations

(Section 3.4.1). The measurement model is defined as follows:

$$h(x_t, v_t) = \begin{pmatrix} x + v_x \\ y + v_y \\ z + v_z \end{pmatrix} \tag{3.37}$$

The Jacobian matrix $H[k]$ is now constant:

$$H_t = \frac{\partial h}{\partial x}|_{x=\hat{x}_t, v=0} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{3.38}$$

Again, the Jacobian V_t is the identity matrix, size N:

$$V_t = \frac{\partial h}{\partial v}|_{x=\hat{x}_t, v=v_t} = I(N) \tag{3.39}$$

Finally the measurement noise is assumed to be constant, so:

$$R = \begin{pmatrix} \sigma^2_{gps_x} & ... & 0 \\ ... & \sigma^2_{gps_y} & ... \\ 0 & ... & \sigma^2_{gps_z} \end{pmatrix} \tag{3.40}$$

If GPS x,y,z calibration data is available it is possible to calculate ‘R’ directly, fully populating the matrix.

Section 3.6 details simulation results of applying this Kalman Filter to improve the positioning of the robot discussed in Section 3.3.3.

3.5.3 Alternative Approaches

The Unscented Kalman Filter [89] reduces the severity of the EKF’s linearisation approximation. Instead of linearising the model, the non-linear model is used to propagate forward a number of discrete samples. The mean and covariance of the estimate are then recovered by averaging the propagated samples. This method accurately captures the true mean and covariance, but is more computationally demanding than the EKF.

The Particle Filter [81] avoids the EKF's linearisation assumption altogether, by sampling the distribution using several 'particles', each of which contains an estimate of the robot's state. At each time step all particles are propagated forward using the non-linear system model. Measurements are used to update a weighting for each particle, favouring those that agree with the measurements. The particles are periodically 'resampled' to spread them over the distribution. If this is not done the majority of particles drift to the low weighted regions. The filter returns the weighted mean of the particle's belief at each time step.

The Particle Filter makes less assumptions than the EKF, so is potentially more accurate, but this comes at the cost of increased computational complexity. The accuracy of the Particle Filter is dependent on the number of particles (samples) and as each particle needs to be propagated forward the filter does not scale well to an increase in the number of states or particles.

Implementation and testing of a Particle Filter fell outwith the scope of this PhD, but this and a detailed comparison with the EKF will take place in future work.

3.6 Kalman Filter Simulation

An EKF was used to refine the positional estimation in Section 3.3.3. The robot global positioning system (Section 2.11.1) was simulated as three beacons positioned at three corners of an imaginary 900mm square held 1m above the work area. A receiver on the robot measured the distance to each beacon. The error in the beacon measurement was characterised by a Gaussian distribution with a variance of $30mm^2$. The three beacons were updated simultaneously at 3Hz.

The EKF simulation results are shown in Figures 3.4 and 3.5. There was a dramatic reduction in positional error. The EKF reduceed the maximum error in robot position from 455mm to 3.35mm. The average positional error was only 1.29mm.

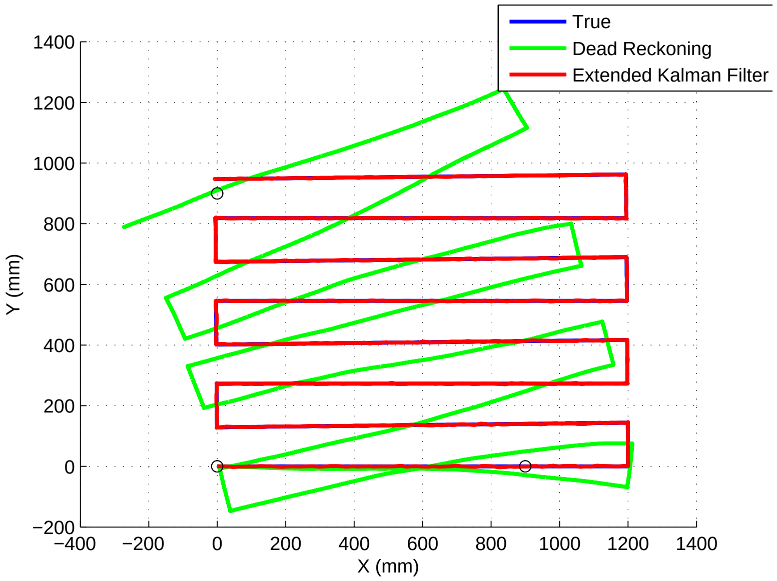


Figure 3.4: Kalman Filter fusing the global and local positional information to improve positional estimate, shown as absolute position.

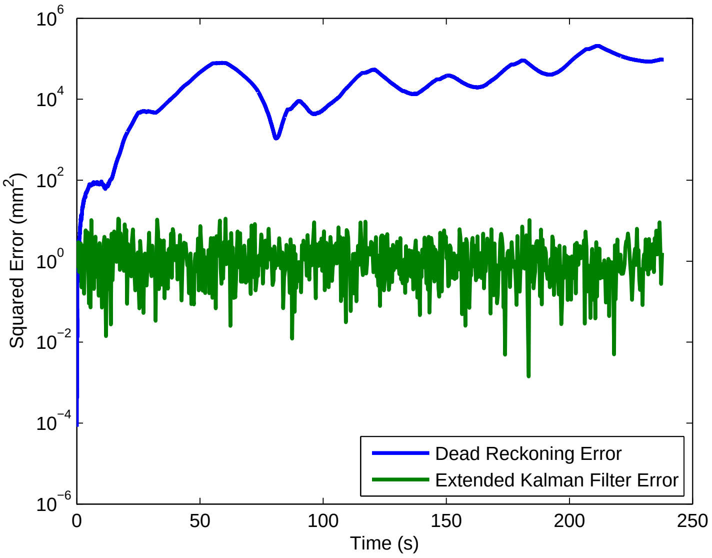


Figure 3.5: A Kalman Filter fusing the global and local positional information to improve positional estimate, shown as squared error.

3.7 Calibration of Measurement Systems

3.7.1 Characterisation and Calibration of Cricket Positioning System

The Cricket Positioning System was validated in 1D by positioning two modules face to face and moving them on a horizontal slider from 100mm to 1900mm in 100mm increments. Plotting the experimental results against the ideal results produced Figure 3.6. The gradient of the line should be unity, but it consistently measured 0.92 irrespective of which modules were used. Interestingly, the Cricket microcontroller is clocked at 7.3728 MHz and the module apparently measures time in microseconds. It appears however, that the developers have used a divide by 8 counter, but neglected to account for the fact that it only pulses at 92% of a microsecond. Future versions of the firmware may fix this, but for the duration of this project the Cricket results were 'fixed' in software. A gain of '1.07' and offset of '-175ms' were applied to all Cricket time-of-flight measurements. Once corrected the variance was measured as 0.152mm, with a approximate Gaussian distribution; this is shown in Figure 3.7.

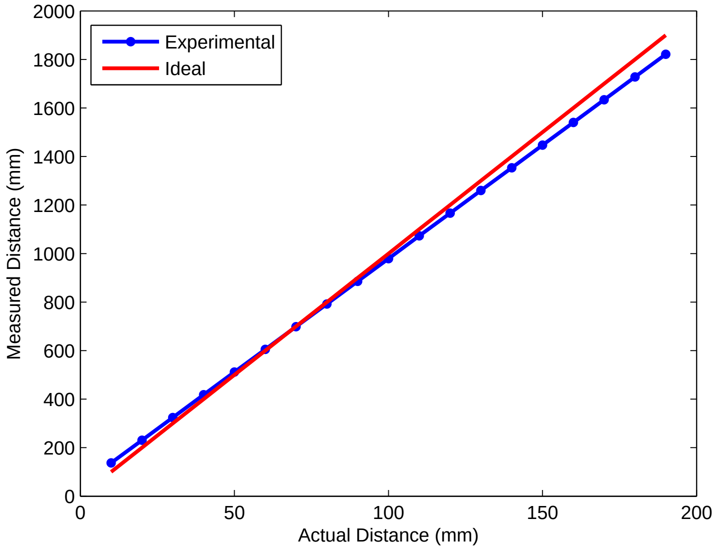


Figure 3.6: Cricket measurements plotted against distance; showing the error in gain that can be attributed to an error in the master clock

The GPS configuration is shown graphically in Figure 2.22, Chapter 2. The beacons are positioned to face the centre of the work area. As documented in Ref. [74] the Cricket measurement has a slight cross-sensitivity to the angle between modules caused by the transducers being non-point sources. This cross sensitivity was mea-

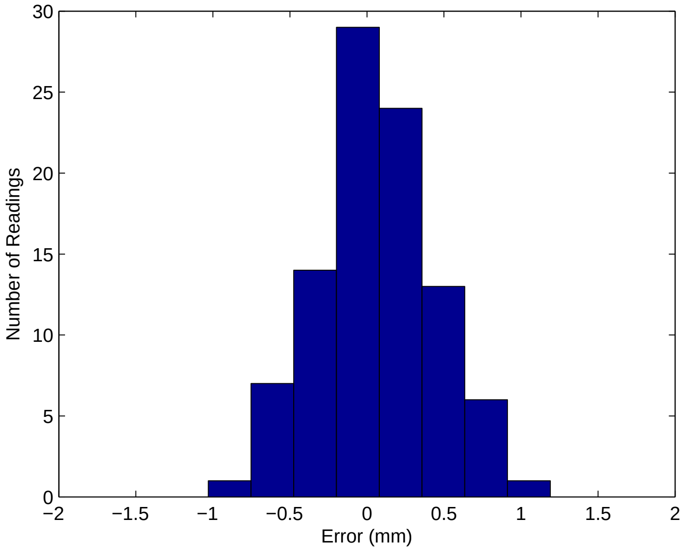


Figure 3.7: 1D Histogram of Cricket in pitch-catch configuration. 5 measurements taken at each position over a range of 2m, with a 100mm increment

sured experimentally in 2D using the arrangement shown in Figure 3.8. The actual distance ‘d’ and angle ‘θ’ were calculated using trigonometry. The error between actual distance and that calculated by Cricket is plotted against angle in Figure 3.9.

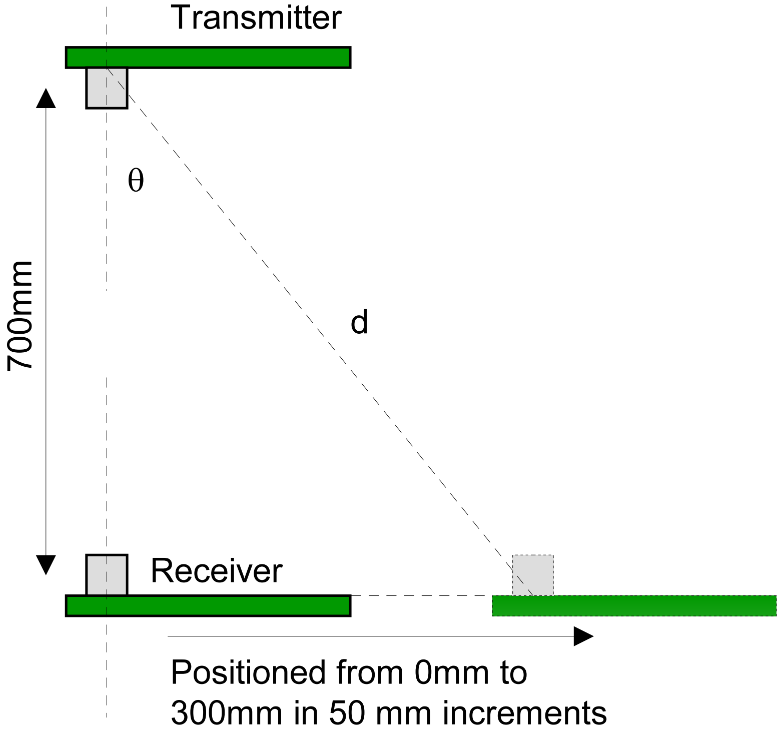


Figure 3.8: Experimental setup to test Cricket sensitivity to angular misalignment

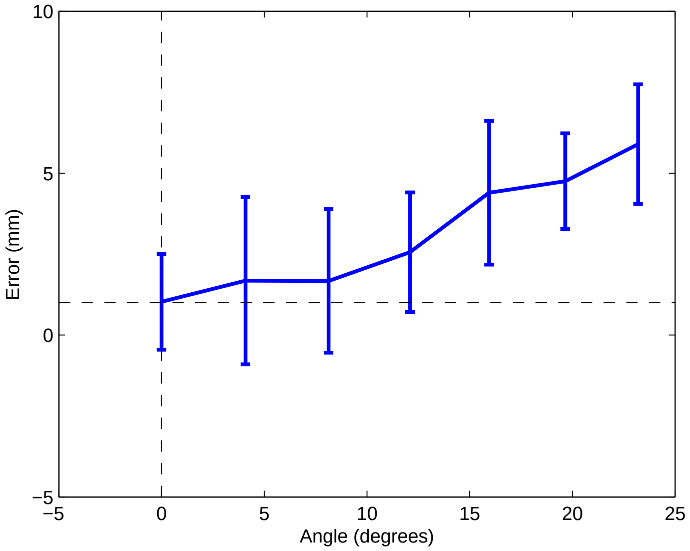


Figure 3.9: Cricket sensitivity to angular misalignment

The actual angular offset between the robot mounted module and the fixed beacons is difficult to define in three dimensions. However, the effect can be clearly seen in a skewing of the Cricket location results shown in Figure 3.10. It was initially suspected that a misplacement of the Cricket modules was the cause of the skewing effect, but the modules were carefully positioned and should be withing 1mm of the desired location. The 2D planar accuracy of the Cricket positioning system was characterised in the following experiment. The robot was positioned in an array of locations. At each location the distance between the robot and each of the three beacons was measured using Cricket. The robot's position was calculated using the triliteration equations (Section 3.4.1). Figure 3.10 shows the actual robot positions in red and the measurement positions in green. The skewing is most evident in the top left hand corner. The beacon locations are shown as blue crosses.

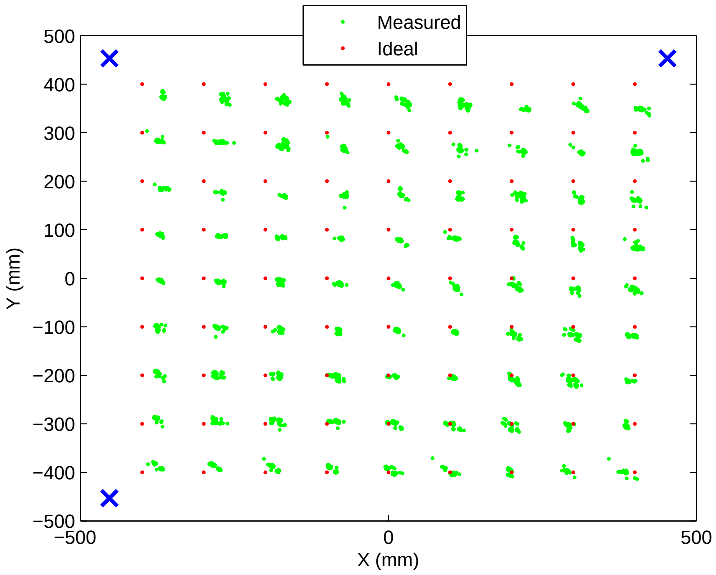


Figure 3.10: Raw data from 2D positioning experiment.
Results were recorded by Rahul Summan

Spatial calibration was investigated as a means of correcting the skewing effect. The spatial calibration uses four (x,y) points, one at each of the corners of the work area. The corners are translated to their ideal positions. All other points are translated using a linear interpolation of the corner translations, in both x and y, resulting in a non-linear transformation.

The algorithm will now be described in more detail. Consider the relative coordinate system of 0 - 1 in both X and Y axis, with the four corners named Top Left (TL), Bottom Left (BL), Top Right (TR) and Bottom Right (BR) as shown in Figure 3.11 - A. The four corners are now measured using Cricket, ideally taking the mean of several measurements to minimise the influence of non-systematic errors. These points are shown in Figure 3.11 - B. Without loss of generality, all points are shifted so that mBL = BL, as shown in Figure 3.11 - C. The linear transform shown in Equation 3.41 is performed, the transformation matrix A is calculated using Equation 3.42.

$$Improved = A \times Measured \tag{3.41}$$

$$A = \begin{pmatrix} mBR_x & mTL_x \\ mBR_y & mTL_y \end{pmatrix}^{-1} \tag{3.42}$$

The result of this linear transformation is shown in Figure 3.11 - D. Finally, the non-linear transformation in Equation 3.43 completes the process, as shown in Figure 3.11 - E. The linear transformation matrix A, and points 'X_{imp}' and 'Y_{imp}' can be calculated offline. Any arbitrary point can then be transformed by performing steps C-E.

$$Calibrated = \left(\begin{array}{cc} \frac{x)}{1-(\frac{y}{y_{imp}})(1-x_{imp}} & \frac{y}{1-(\frac{x}{x_{imp}})(1-y_{imp})} \end{array} \right) \tag{3.43}$$

where:

$$\begin{pmatrix} x_{imp} \\ y_{imp} \end{pmatrix} = A \times mTR \tag{3.44}$$

Calibration data was extracted from Figure 3.10 as the mean of each of the four corner clusters. Spatial calibration was then performed on the remaining data to produce Figure 3.12.

Histograms were generated for all X and Y calibrated points, subtracting the ideal coordinates from the measured coordinates. The histogram before calibration is shown in Figure 3.13. After calibration the mean shifts to zero and the variance is reduced. The histogram can be shown as a 2D colour plot, as shown in Figure 3.15. From this data, the covariance matrix for the Cricket measurement system, in the defined work area, was calculated as:

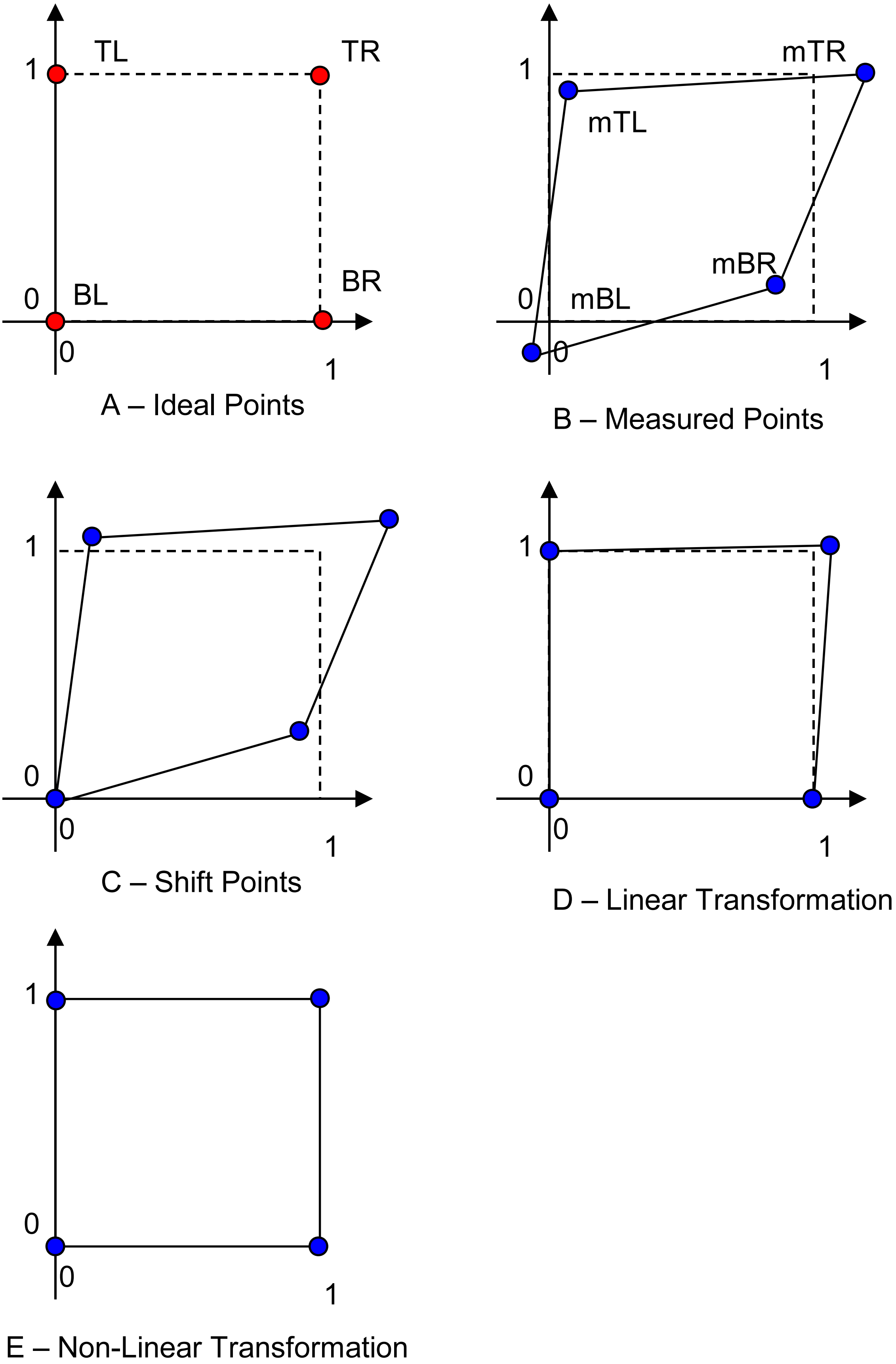


Figure 3.11: Spatial calibration procedure for Cricket Indoor Positioning System

$$Covariance = \begin{pmatrix} 31.4 & -16.1 \\ -16.1 & 27.8 \end{pmatrix}$$

The variance is 31.4 mm^2 in the X direction and 27.8 mm^2 in the Y direction. This covariance matrix can be used directly in the Kalman Filter measurement update stage,as the measurement covariance matrix 'R'.

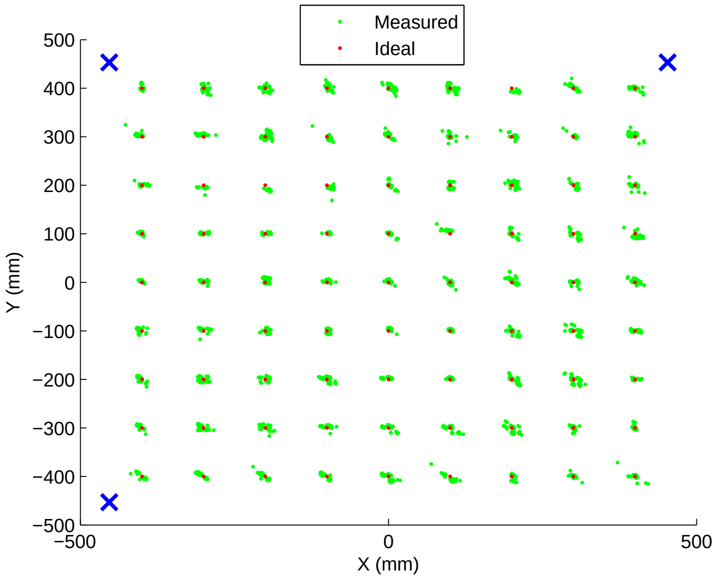


Figure 3.12: 2D positioning experiment after calibration . Compare with Figure 3.10

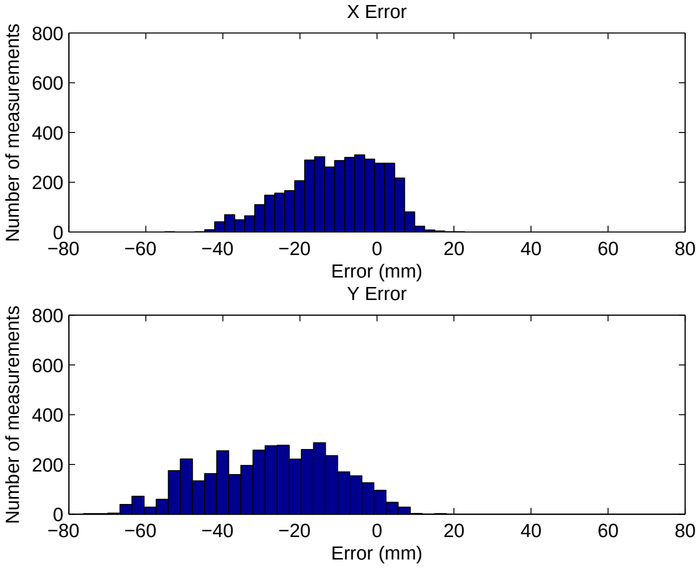


Figure 3.13: Histogram of Cricket X/Y data before calibration

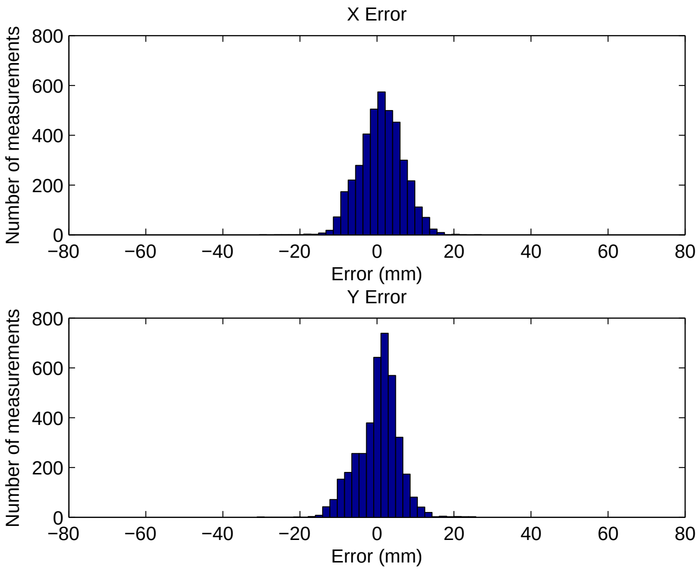


Figure 3.14: Histogram of Cricket X/Y data after calibration

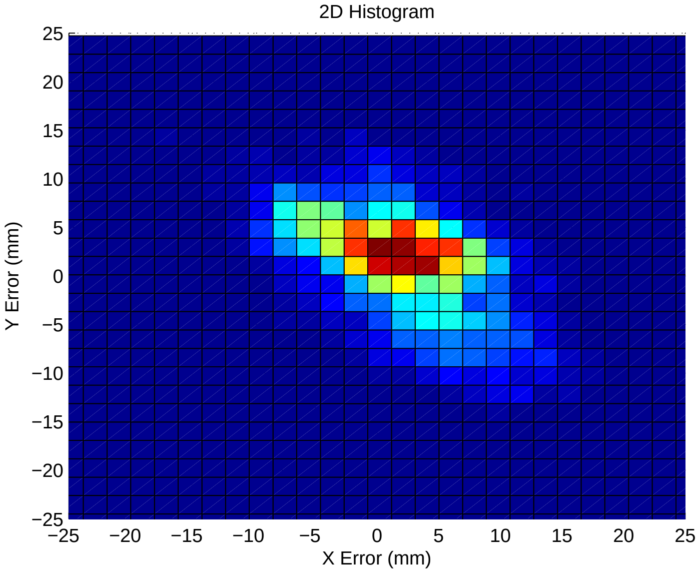


Figure 3.15: 2D histogram of calibrated Cricket positional results

3.8 Implementation of the Kalman Filter

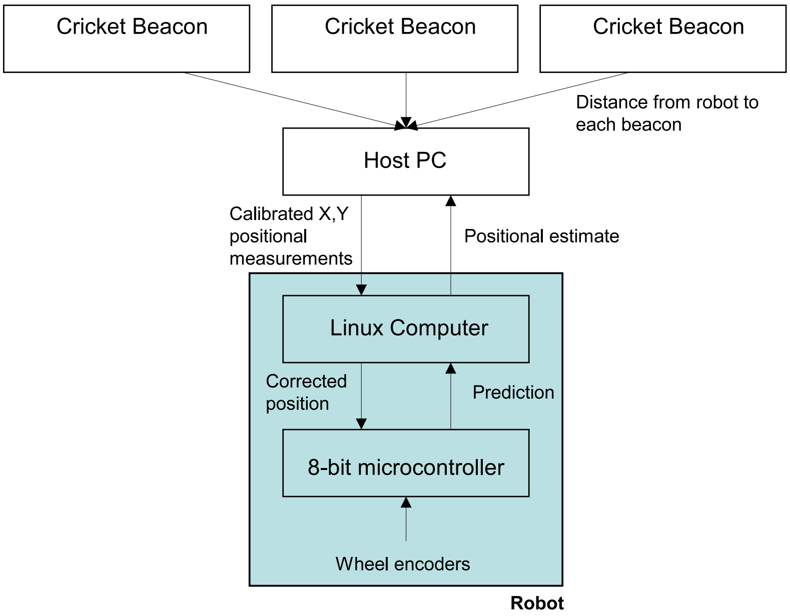


Figure 3.16: Block diagram showing the extended Kalman filter implementation

Figure 3.16 shows a block diagram of the EKF implementation. For convenience, the majority of the processing is performed on the embedded Linux computer. The EKF is evaluated when a new Cricket measurement becomes available. The Cricket positioning system was configured to have three passive beacons, as discussed in Section 2.11.1, Chapter 2. In this configuration, the host PC receives three simultaneous distance measurements at approximately 3Hz. X,Y coordinates were calculated using the triliteration equations discussed in Section 3.4.1. The X,Y coordinates were spatially calibrated using the algorithm discussed in Section 3.7.1. The coordinates were then transmitted to the robot over WiFi. The Cricket measurement covariance matrix calculated in 3.7.1 was used directly in the EKF.

The wheel encoder readings were processed by the 8-bit microcontroller. To save excessive transmission of the encoder readings between processors, the prediction stage was performed on the 8-bit microcontroller. The current prediction is requested by the Linux computer each time it processes the EKF. The encoder variances were tuned experimentally. On completion of an EKF update, the X,Y correction is transmitted back to the 8-bit microcontroller, which updates its internal record of the robot's position. If the Cricket measurements become unavailable, the prediction stage is no longer corrected, resulting in regular dead-reckoning.

3.8.1 UMBmark

UMBmark [84] is a calibration procedure used to tune the mechanical properties of a differential drive robot. It reduces the dominant systematic errors, improving the accuracy of the dead-reckoning model discussed in Section 3.3.2. Borestein et al [84], state that two dominant systematic errors are unequal wheel diameters and uncertainty about the wheelbase. The error due to unequal wheel diameters, D_l (left) and D_r (right), is defined as:

$$E_d = \frac{D_r}{D_l} \tag{3.45}$$

The uncertainty in wheelbase, 'b', is defined as:

$$E_b = \frac{b_{actual}}{b_{nominal}} \tag{3.46}$$

UMBmark does not correct for inaccuracies in the average wheel diameter (scaling error), but this can be easily measured by programming the robot to go in a straight line for a set distance, l, and measuring the inaccuracy in distance travelled.

$$E_s = \frac{l_{calc}}{l_{nominal}} \tag{3.47}$$

The compensated diameter, Dn*, is calculated as:

$$Dn^* = E_s D_n \tag{3.48}$$

The effects of E_b and E_d on robot positioning over a square path are shown in Figure 3.17.

UMBmark was performed by driving the robot in a pre-programmed square path of length L in both clockwise and anticlockwise directions, each time measuring the error in robot destination (ϵ_x and ϵ_y). Performing the measurement in both directions ensures the errors are not concealed when mutually compensating. Non-systematic errors are minimised by taking the average of several runs.

The following geometric relationships were calculated:

$$\alpha = \frac{\epsilon_{x,cw} + \epsilon_{x,ccw}}{-4L} \frac{180^o}{\pi} \tag{3.49}$$

$$\beta = \frac{\epsilon_{y,cw} - \epsilon_{y,ccw}}{-4L} \frac{180^o}{\pi} \tag{3.50}$$

The error in wheel base E_b is calculated as:

$$E_b = \frac{90^o}{90^o - \alpha} \tag{3.51}$$

The actual wheelbase is calculated as:

$$b_{actual} = E_b b_{nominal} \tag{3.52}$$

The curvature of the path is defined as:

$$R = \frac{L}{2sin(\frac{\beta}{2})} \tag{3.53}$$

The error due to unequal wheel diameters was calculated as:

$$E_b = \frac{R + \frac{b}{2}}{R - \frac{b}{2}} \tag{3.54}$$

The correction factors for left and right wheels can be calculated as:

$$c_l = \frac{2}{E_d + 1} \quad c_r = \frac{2}{\frac{1}{E_d} + 1} \tag{3.55}$$

UMBmark was preformed with square length of 600mm. The nominal wheel base was 72mm with a nominal radius of 41.5mm. UMBmark revealed an actual wheel base of 73.1mm, a 0.1% increase in D_l and a 0.1% decrease in D_r. The positional results are shown in Table 3.1.

Before UMBmark	**CW**	**CCW**
Average error in destination ϵ (mm) Standard Deviation of ϵ (mm)	(-47.7,-37.2) (3.19,4.6)	(-10.3,-30.3) (2.6 5.8)
After UMBmark		
Average error in destination ϵ (mm) Standard Deviation of ϵ (mm)	(7.3,6.9) (5.1,1.7)	(-2.5,11.6) (4.3,2.8)

Table 3.1: UMBMark results

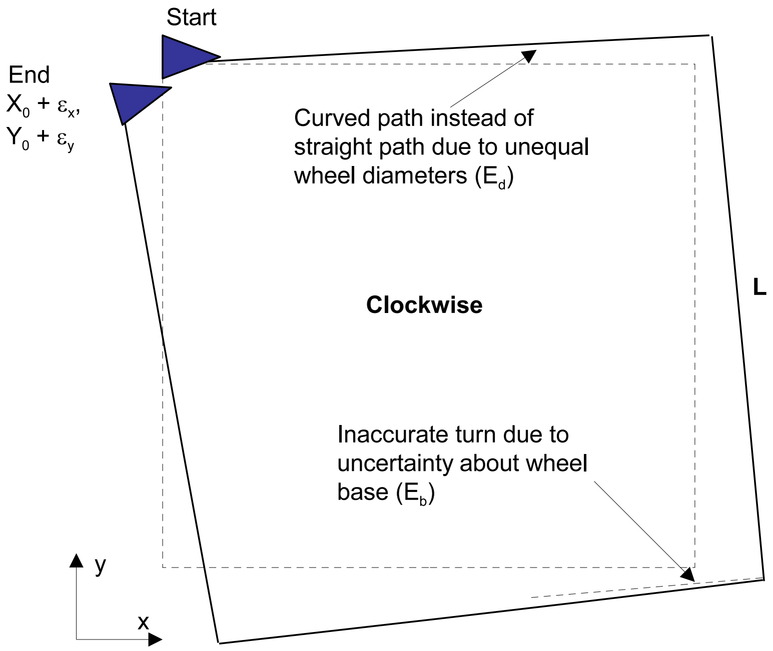

Start
End
$X_0 + \varepsilon_x$,
$Y_0 + \varepsilon_y$
Curved path instead of straight path due to unequal wheel diameters (E_d)
L
Clockwise
Inaccurate turn due to uncertainty about wheel base (E_b)
y
x

Figure 3.17: Two dominant systematic errors affecting dead reckoning

3.9 Robot Control

A fundamental task in robotic NDE is the ability to raster scan the specimen, thus providing complete coverage of the specimen. The scan can be broken into a series of 'move to coordinate' commands (See Section 2.9.4, Chapter 2) each of which consists of a rotation to face the goal and a straight line segment from the current location to the goal. A controller is required to maintain accurate control in the presence of any disturbances.

A heading controller, designed to minimise the error in the robot's path, was configured across the two drive motors. In its simplest form the controller corrects for the angular error in the robot's heading shown in Figure 3.18. The action of the controller is to increase the set speed of one wheel and reduce the set speed of the other wheel until the error in heading is eliminated, i.e.

$$w_l = w_l + K_{heading}E$$

$$w_r = w_r - K_{heading}E \tag{3.56}$$

Where E is the heading error in degrees and $K_{heading}$ is the controller gain. A simulated example where the robot's position is off by 5^o is shown in Figure 3.19. The limitation of this controller is that once the robot leaves the desired path it does not meet it again until the destination, this is shown in Figure 3.19. It would be preferable for the robot to head back to the desired path, rather than directly to the destination.

This was achieved with an extended version of the controller that adds a second cost term that measures the distance between the robot's current position and the ideal path shown in Figure 3.18. This extended controller is defined as:

$$w_l = w_l + K_{heading}E_{heading} + K_{line}E_{line}$$

$$w_r = w_r - K_{heading}E_{heading} - K_{line}E_{line} \tag{3.57}$$

Where K_{line} is an additional controller gain and E_{line} is the shortest distance between the robot and the ideal path. The improved response to a 5^o error in heading is shown in Figure 3.20. Here, the robot rapidly returns to the intended path. Finally, an additional logic term was added to ensure that the robot does not over steer towards the line. This was achieved by limiting θ_l, shown in Figure 3.18. The logic term only activates the controller when θ_l is less than a threshold, the threshold was set at 20^o. The controller can be tuned by modifying the two gains, higher values tend to lead to overshoot and instability, low values make the controller less effective.

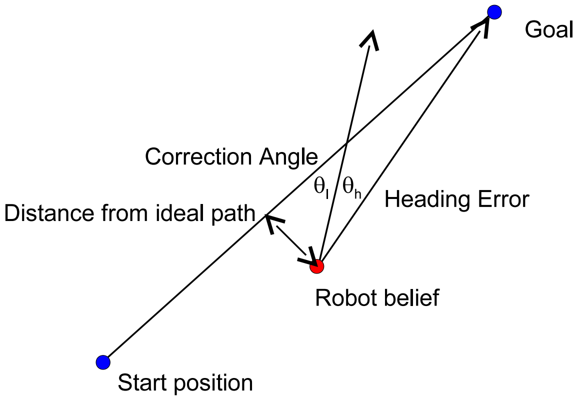


Figure 3.18: Diagram showing the two significant positioning errors Firstly the error in heading between the current heading and the required heading to reach the goal. Secondly, the distance between the current position and the ideal path.

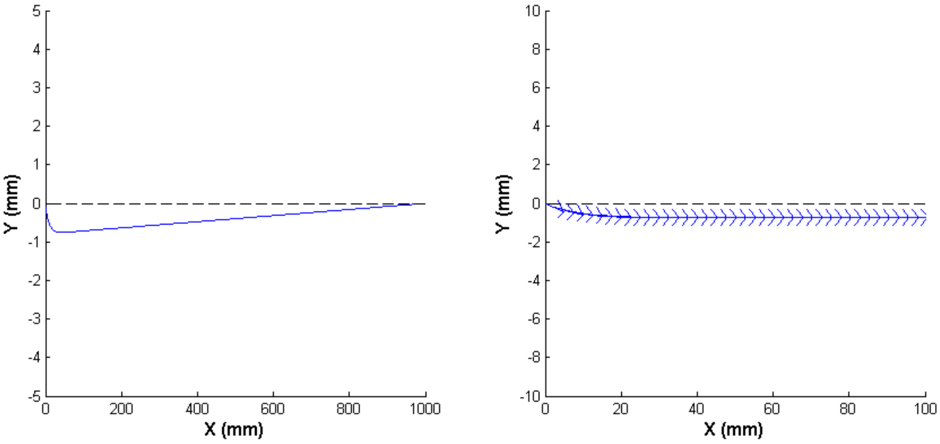


Figure 3.19: Heading correction control. The figure on the right includes orientation and focuses on the initial 10% of the path

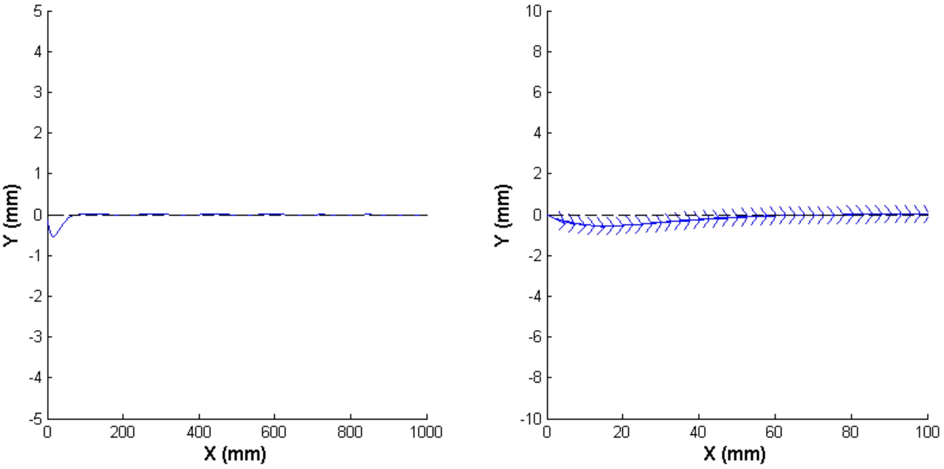


Figure 3.20: Straight line controller. The figure on the right includes orientation and focuses on the initial 10% of the path

3.10 Experimental Results

The following section discusses positioning of the robot for several different paths. The robot's 'exact' location is recorded using the Visual Positioning System discussed in Section 2.11.2, Chapter 2. The robot's belief is taken directly from the embedded EKF.

3.10.1 Repeating Square Path

The first path was a square with 600 mm sides. It was repeated five times. In the first experiment the robot used dead reckoning, the result is shown in Figure 3.21. As expected, there is a growth in positional inaccuracy. It may be possible to improve the accuracy by further refinement of tolerances in the robot's mechanics, but this would not fix the intrinsic problem of integrating uncertainty.

Figure 3.22 shows the same path, enhanced using the EKF. There is a dramatic improvement in positional accuracy. The robot is always withing 20 mm of the desired location and within 10mm of it's own belief. Interestingly, the robot follows a fairly consistent path with a maximum spread of 10mm (for 5 repetitions of the square path), so at the worst region every point on all five repetitions is off by over 10mm. Initially, it was suspected that the Cricket measurements may not have been perfectly calibrated, however Figure 3.23 shows that they are in relatively good agreement with the visual positioning system.

The subtle, but important difference between the simulation discussed in Section 3.6 and these experiments is that in the simulation, the robot's drive signals are perfect, it follows the ideal path and the EKF only tracks it. In the experiments, the robot's motor drive signals are based on its own belief of position which is generated by the EKF. The slight curvature that can be seen in the straight sections is a combination of the EKF refining the location estimate and the on board controller trying to keep the robot's belief on the desired path.

The EKF is configured with the variance of each measurement system; the variance of the wheel encoder measurements and the variance of the Cricket indoor positioning system. These two values control how much confidence the filter has in each type of measurement. There is a notable lag between the robot's belief deviating from the Cricket measurements and the EKF correcting the belief. This suggests the EKF was putting too much faith in the wheel encoders and not enough in the Cricket measurements. Although it was straightforward to measure the variance of the Cricket positioning system, as is shown in Section 3.7.1, there is no straightforward way to measure the variance of the wheel encoders. Instead this value was tuned experimentally. Increasing the EKF's notion of the variance of the wheel encoders, hence telling the filter that they are less accurate, causes better correlation between Cricket and the robot's belief in X and Y, but since Cricket does not track orientation, the robot's notion of orientation deteriorated. This led to erratic results with greater curvature and increased the maximum deviation between the robot's measured and desired locations.

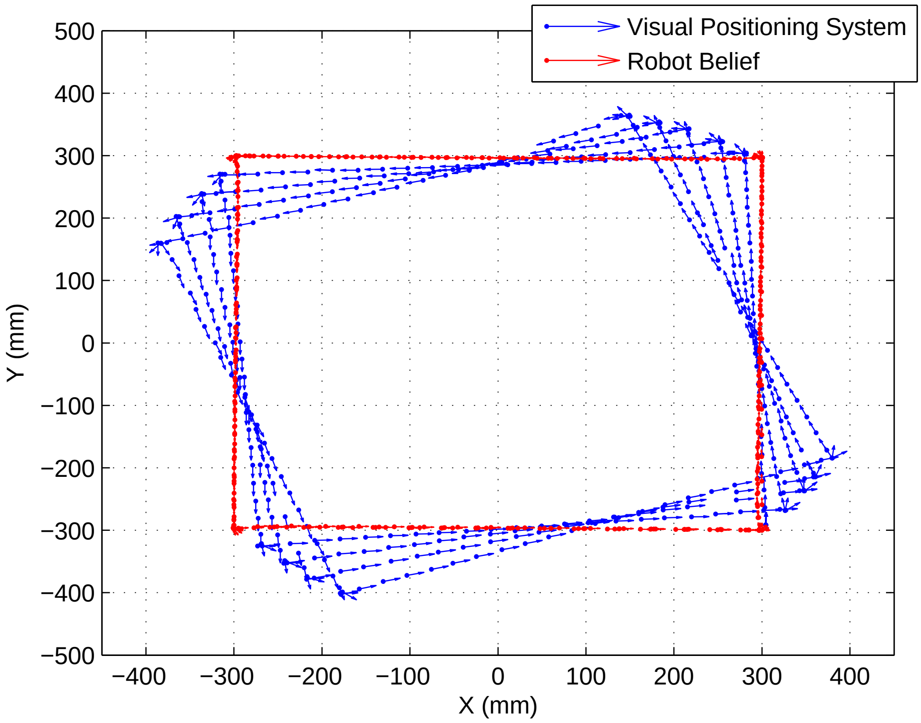


Figure 3.21: Robot following repeating square path using only dead reckoning.

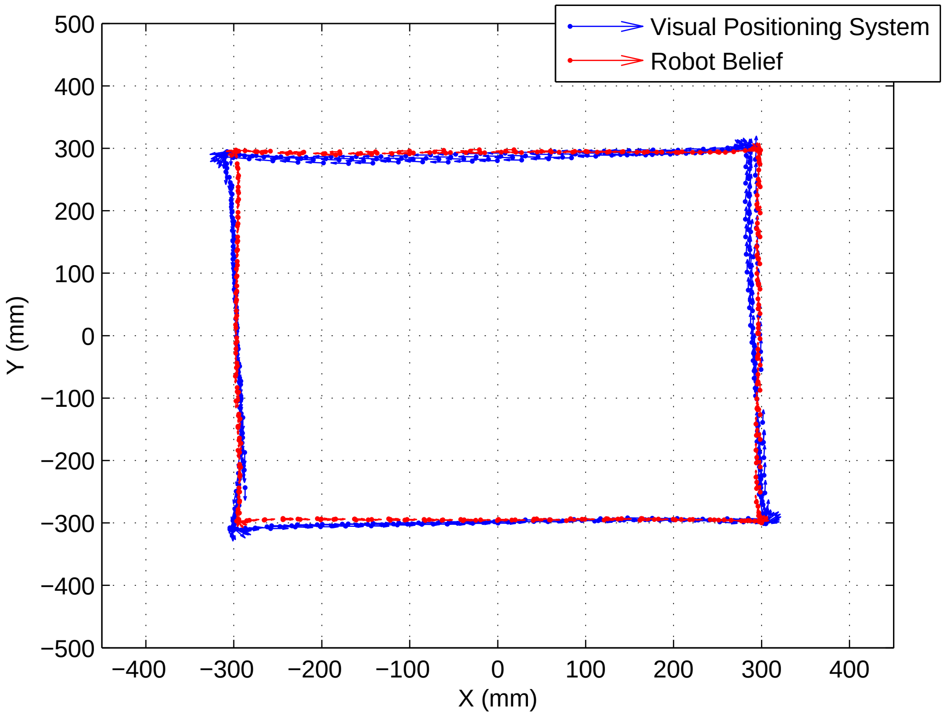


Figure 3.22: Robot following repeating square path using Kalman filter for positional tracking.

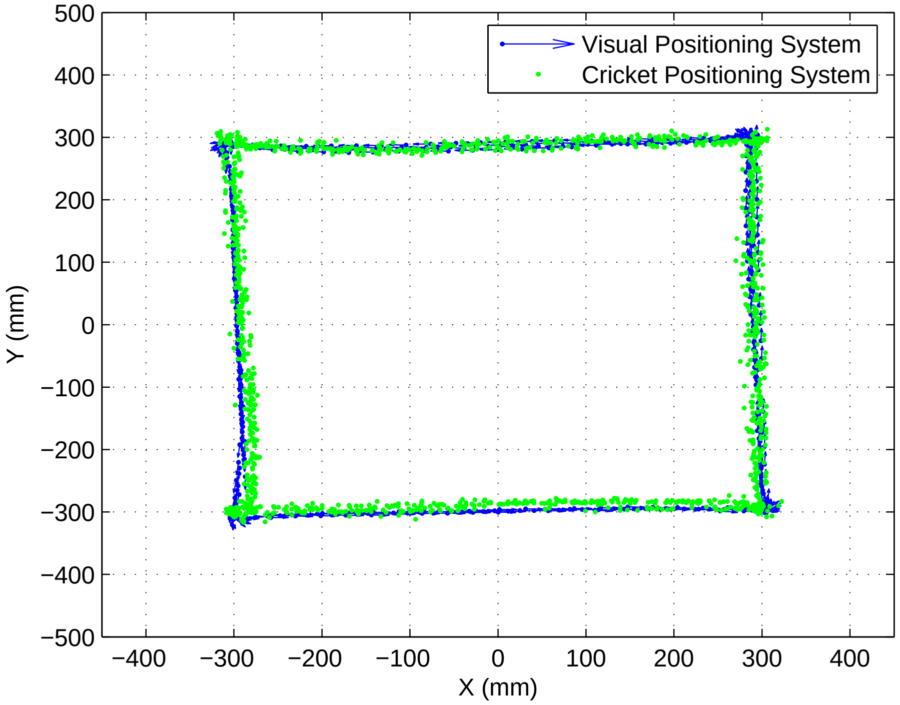


Figure 3.23: Cricket GPS measurements used in Figure 3.22
The plot shows the X,Y measurements from the Cricket positioning system

3.10.2 Raster Scan

The second scan path was a raster scan 600mm wide, with 100mm spacing between scan lines. Figure 3.25 shows the scan relying on dead reckoning. The integrating error in position is again evident. Figure 3.26 shows the scan using the EKF. The positional accuracy is inferior to the square path with the robot drifting to up to 32mm away from the required position. This appeared to be due to the high density of turns in the scan pattern. The 100mm vertical sections did not give the filter sufficient time to refine its orientation belief before the next turn. To address this, the scan pattern was modified to increase the length of the vertical sections to 400mm on the left and 300mm on the right, sending the robot in a 'looping' pattern that yielded 100mm spacing in the horizontal scan lines. The modified scan pattern is shown in Figure 3.24

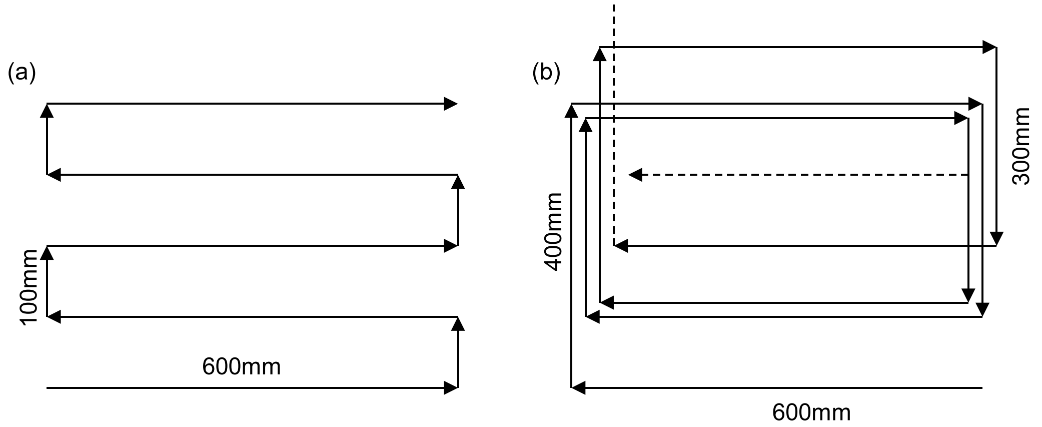


Figure 3.24: (a) Shows the original scan path, however the short 100mm sections did not give the robot time to recover from a turn. (b) The modified path to increase the minimum length of straight line sections.

The result is shown in Figure 3.27. It shows a considerable improvement in accuracy, reducing the maximum error to less than 10mm, but increasing the scan length by approximately 20%.

Figure 3.28 shows the modified raster scan with a more practical 33.3mm between horizontal scan lines. The maximum error between robot belief and actual position is less than 10mm. There is fairly uniform surface coverage.

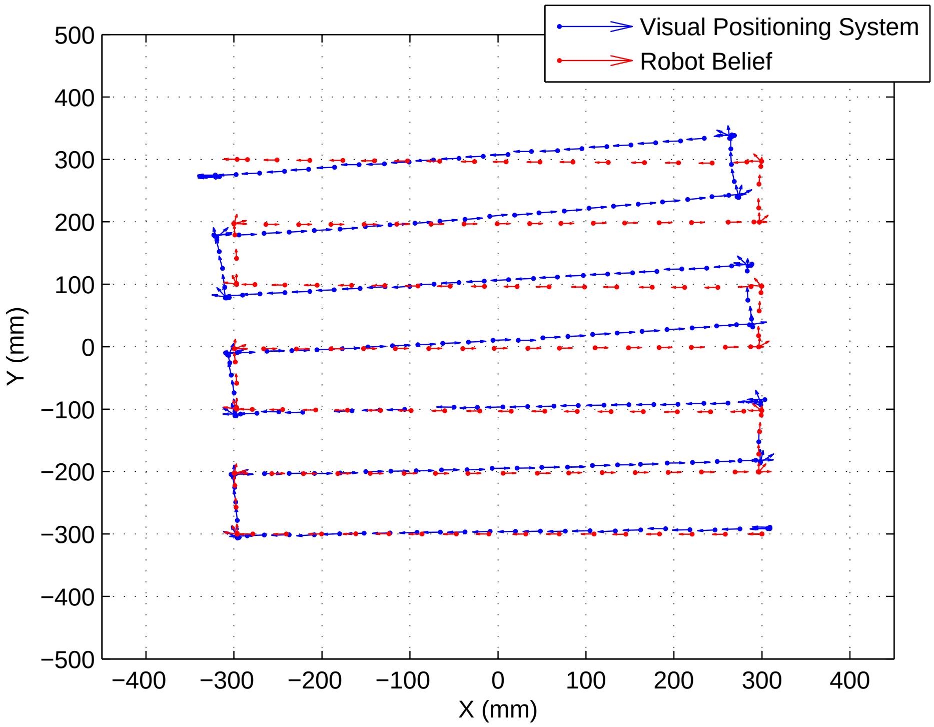


Figure 3.25: Robot following raster scan using only dead reckoning

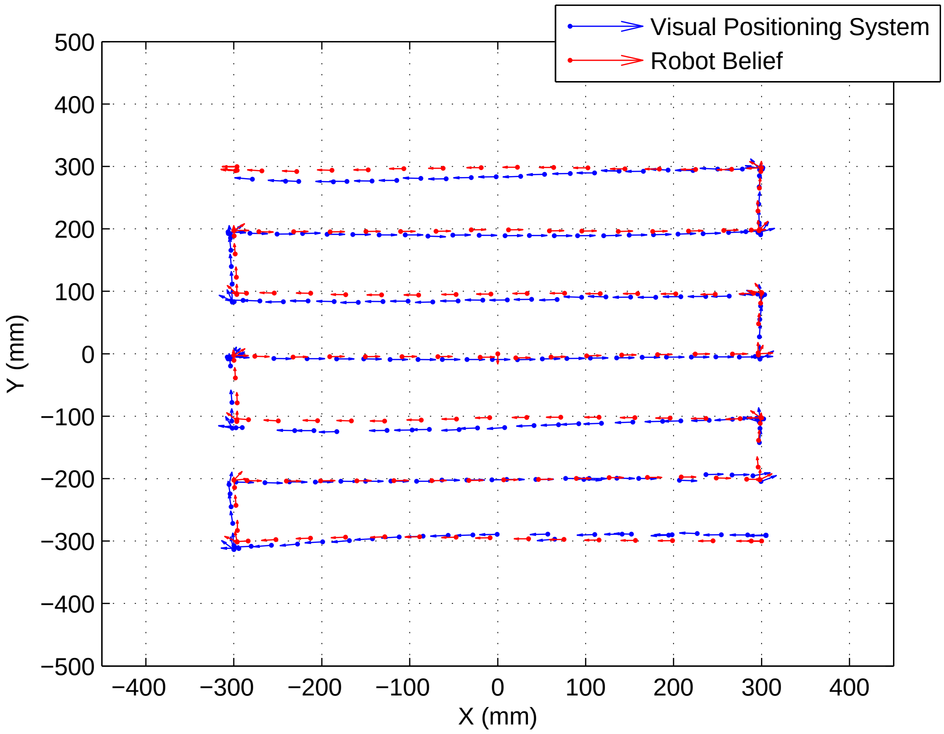


Figure 3.26: Robot following raster scan using the EKF

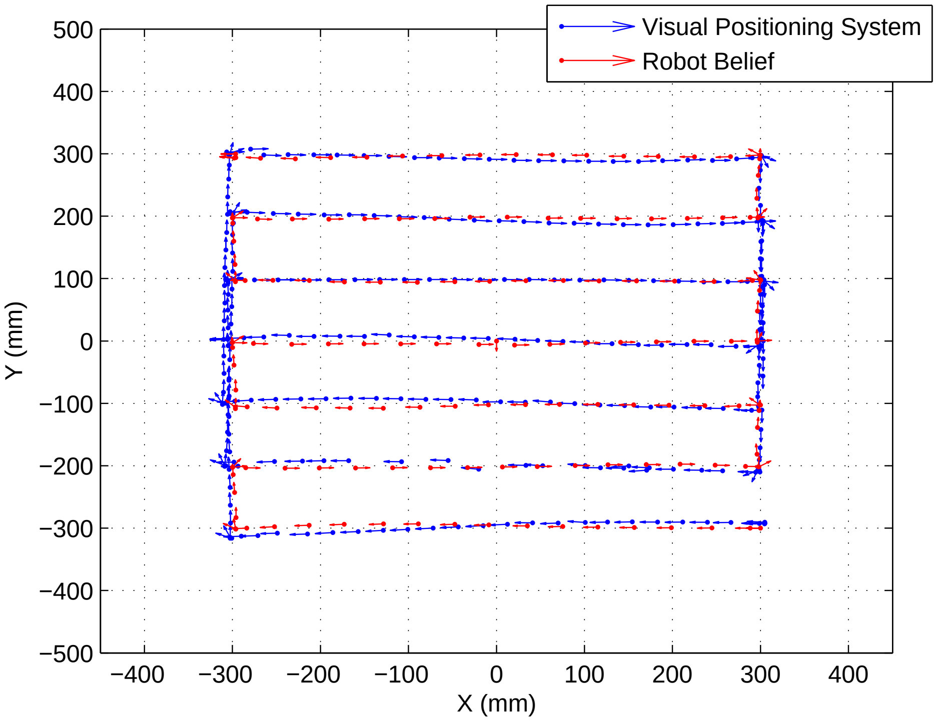


Figure 3.27: Robot following modified raster scan using the EKF

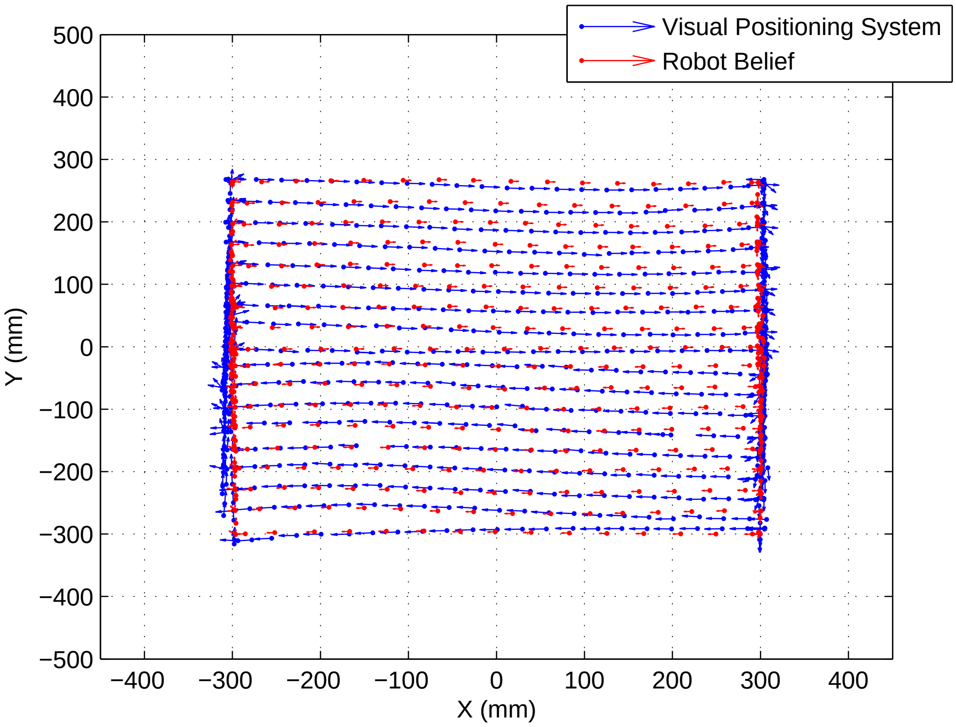


Figure 3.28: Robot following fine modified raster scan using the EKF

3.11 Conclusions and Future Work

This Chapter has detailed the successful application of an EKF to the differential drive robot discussed in Chapter 2. In a typical experimental raster scan the positional estimate has been demonstrated to be within 10mm of the actual robot position. Several steps were performed to achieve this:

1. UMBmark was applied to tune the robot's system model - Section 3.8.1.
2. The Cricket Global Positioning System was spatially calibrated - Section 3.7.1. The validation results were used to generate R_k, the measurement noise error covariance matrix - Section 3.5.
3. A heading correction controller was created to continually drives the robot toward the intended path - Section 3.9.
4. The Visual Positioning System was used to characterise the performance tracking - Section 2.11.2, Chapter 2.
5. The EKF was tuned by adjusting the wheel encoder measurement variance.

The static model used in the prediction stage of the EKF (Section 3.3.2) appears to be a limiting factor in accurate positional estimation. Although UMBmark 3.8.1 has improved the correlation between the static model and the hardware, the results are not ideal. The consistent angular offset between iterations of the square path in Figure 3.21 suggest a systemic error leading to over turning in each 90^o turn. In addition to this, the EKF assumes that the error from the wheel encoders is Gaussian. This approximation will break down if there is any wheel slip, something that is likely to occur when the robot is turning. Additionally, the filter does not include an error term for small errors in the static model, such as a slight difference in the distance between each wheel and the centre point. Since the wheels are in contact with the surface over a finite width (1mm), the exact wheel base is difficult to determine.

The Cricket positioning system has limited measurement accuracy (variance of $30mm^2$) with a relatively slow refresh rate of 3Hz. Improving either the accuracy, or refresh rate will lead to an improvement in the EKF's positional estimate. In simulation, reducing the Cricket variance from $30mm^2$ to $3mm^2$ reduced the peak error from 3.35mm to 1.21mm. Similarly, increasing the refresh rate from 3Hz to

10Hz (keeping variance at $30mm^2$), reduced the peak positional error to 1.8mm. A major research effort in the Centre for Ultrasonic Engineering at the University of Strathclyde is aiming to create an ultrasonic positioning system that is significantly more accurate than Cricket. The system emits ultrasonic chirps and uses a phase measurement to calculate distance. Initial results have demonstrated 1D distance accuracies of 0.06mm. Integrating this into the robot platform will improve future positional performance.

The robot was equipped with a gyroscope for relative angular measurement. Unfortunately, the gyroscope was discontinued by the manufacturer during the early part of this work, so was not used in the EKF study. Integrating an accurate gyroscope into the EKF should enhance results. A global orientation measurement would improve the positional estimate further by obviating the need to rely on the relative measurements of robot orientation.

The EKF linearises the system model, leading to a degree of inaccuracy. The severity of this is dependent on the level of non-linearity in the system model. A Particle Filter does not require system model liniarisation, but it is computationally more expensive. Future work will directly compare the accuracy of both techniques when applied to the miniature robotic vehicle platform.

Vision based positioning systems such as the Vicon MX T160 [77] are able to track objects with accuracy approaching 0.05mm at 120Hz. The Vicon system tracks passive spherical targets with a diameter of 3mm upward. By mounting several targets on a single robot it is possible to track its orientation. The positional measurements from a system of this accuracy could be used directly for positional estimation, obviating the need for a Kalman Filter. However these systems are expensive (in the order of £100,000) and require a reasonable amount of infrastructure, making them unsuitable for some industrial applications. Passive optical systems are also sensitive to ambient light conditions, causing them to fail in some situations. Vision based systems with active LED based targets, reduce the effects of ambient light, but require a wire between the targets and positioning system. This makes them unsuitable for the miniature robotic vehicle platform. Strathclyde University has recently completed the purchase of a Vicon MX T160 system which will be evaluated experimentally in the near future.

The ultimate objective of robot positioning is accurate positioning in an unknown environment, without the need for a global positioning system. The **S**imultaneous

Localisation **A**nd **M**apping (SLAM) algorithm [81] aims to achieve this. The algorithm projects sensor data onto a map which is then used for positioning. If the same feature is detected more than once, SLAM attempts to improve the quality of the map, and position estimate, by overlapping the features. Consider the simple example of a robot instructed to follow a circular path. As the robot progresses along the path, a build up of integral errors causes the accuracy of the map and positional estimate to degrade. However, as the robot approaches the end of the path, it detects features that are already in its internal map (detected at the start). SLAM uses this information to amend the map so that the start and end of the path overlap. This improves the quality of the map and positional estimate. SLAM systems typically make use of cameras and laser range finders to scan the environment and extract features. Future work at the University of Strathclyde will attempt to implement SLAM on the miniature robot platform.

Chapter 4

A Non-Contact Ultrasonics Payload

4.1 Introduction

Non-contact ultrasonics offers significant potential for wireless robotic inspection. This chapter details work to embed the existing air-coupled Lamb wave inspection hardware, shown in Figure 4.1, into the miniature robotic platform. This payload is the first truly portable non-contact Lamb wave inspection system.

The Chapter begins with an overview of ultrasound wave propagation, an introduction into the use of ultrasound for NDE and an overview of ultrasonic transducer technologies. The chapter then describes the design of the new payload in detail, before finishing with an evaluation of the system.

4.2 Principles of Ultrasound

Ultrasound, by definition is sound with a frequency too high to be heard by the human ear. This is normally above 20 kHz. Sound is the transmission of a mechanical disturbance from the equilibrium of the particles in a elastic material. It propagates in the form of waves which can undergo interference, diffraction, reflection and refraction. Depending on the type and geometry of the material, sound can propagate in several modes including longitudinal, shear, surface waves and Lamb waves.

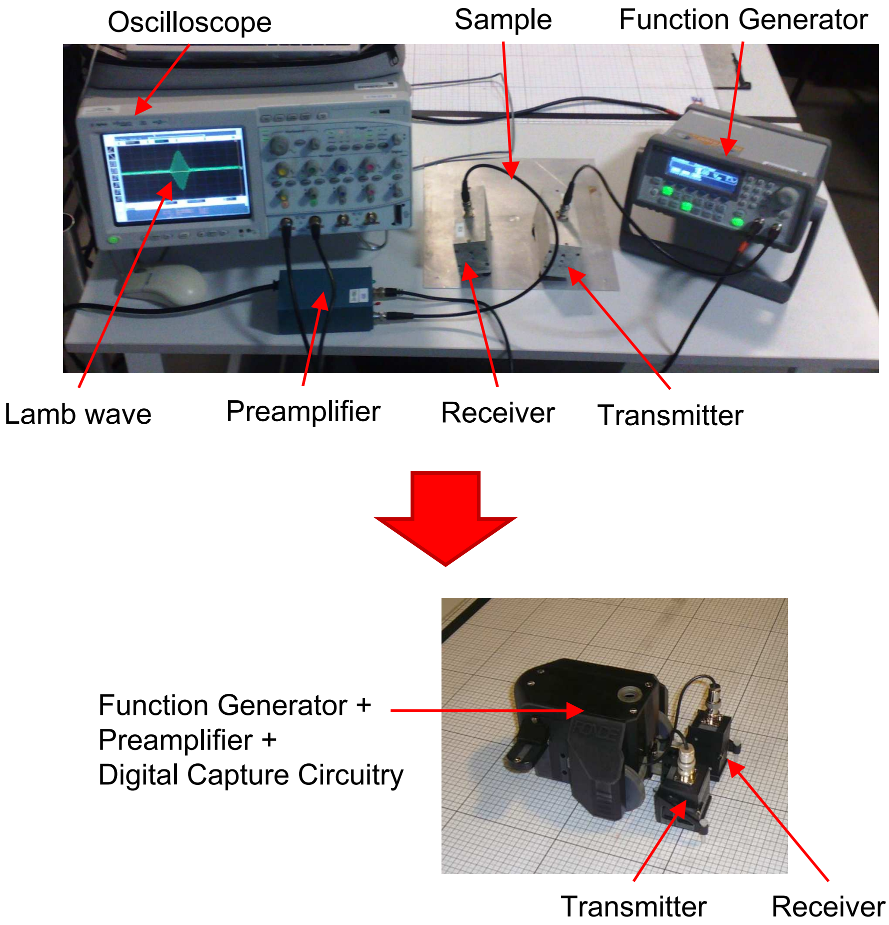


Figure 4.1: Top: A traditional Lamb wave inspection experiment.
Bottom: Embedded ultrasonic payload

4.2.1 Modes of Ultrasonic Propagation

In a longitudinal wave the particle motion is parallel to the direction of propagation, resulting in a wave that compresses and dilates the material. Longitudinal waves are often referred to as compression waves, they can exist in solids, liquids and gases.

In a transverse or shear wave the particle motion is perpendicular to the direction of wave propagation. A unique characteristic of shear waves is the absence of regions of compression and rarefaction seen in other types of wave. For most practical purposes, shear waves can only exist in solids.

Rayleigh waves [9] are a surface wave that propagates along the boundary between a solid and a vacuum or sufficiently rarefied medium, such as air. The particles on the surface exhibit elliptical motion which is localised to the surface and decays exponentially as the depth increases. The vibration amplitude is over 90% lower at a depth of one wavelength.

When the material is sufficiently thin to allow a Rayleigh wave to propagate the entire way through, such as in plates, the surface wave becomes a Lamb wave [9]. Lamb waves are a combination of shear and longitudinal components that phase match at the material boundaries. This combination can be realised in an infinite number of symmetrical and antisymmetrical modes, generally referred to as A_0, A_1, A_2 ... and S_0, S_1, S_2 ... respectively. The fundamental modes, A_0 and S_0, are the only modes to exist at all frequencies-thickness products. In practice, they are easier to excite and therefore typically carry more energy than the higher order modes.

The fundamental symmetrical mode can be thought of as the propagation of a series of ‘bulges’, with a pure compression wave in the centre. This is shown in Figure 4.2. The fundamental antisymmetric modes consists of a propagation of localised plate bending as shown in Figure 4.2. The motion at the centre is pure shear.

Lamb wave propagation velocity is dependent on the wave’s frequency multiplied by the sample thickness. This is generally referred to as the frequency-thickness product. This relationship means that the frequency components of a Lamb wave travelling in a plate with constant thickness may move at different velocities. For example, if the wave originates as an impulse, different frequencies components will spread out, leading to an elongation of the wave packet. This is referred to as dispersion. Figure 4.2 shows the Lamb wave dispersion curve for 1 mm aluminium; it plots the phase velocity against frequency for each mode. The plot was created using the software program Disperse [90].

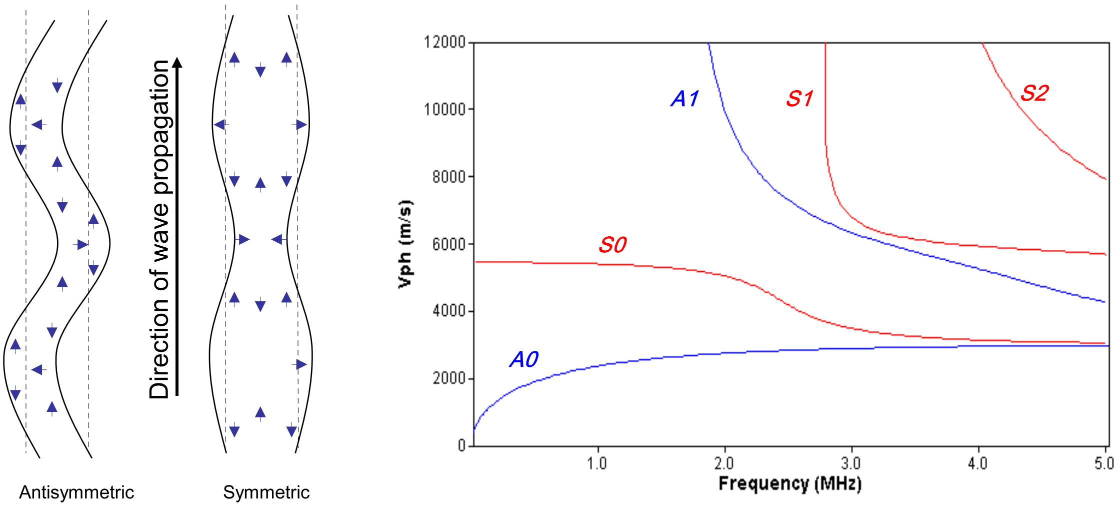


Figure 4.2: Diagram showing Lamb waves modes and a dispersion plot (1 mm aluminium)

4.2.1.1 Mode Conversion

It is possible to convert one type of sound wave to another in a process known as mode conversion. Using this phenomenon, it is possible to generate Lamb waves from a longitudinal excitation, which is of critical importance to this work.

Consider the boundary between a liquid or gas and a solid, shown in Figure 4.3. When a longitudinal wave is directed towards the boundary at an angle from the normal, mode conversion occurs. The refracted waves travel at different angles and speeds which can be calculated using Snell's law [9], Equation 4.1.

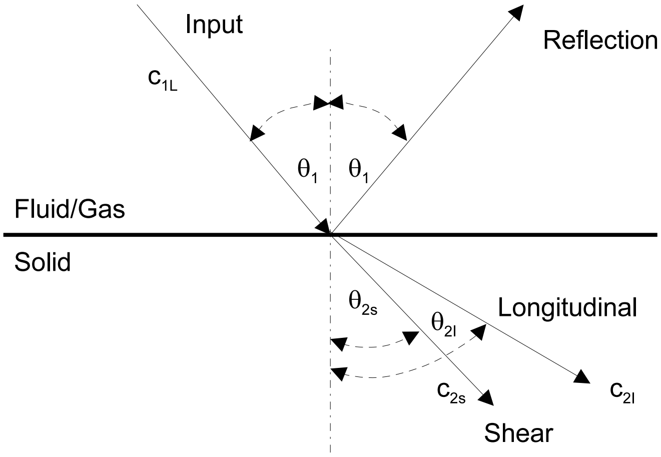


Figure 4.3: Mode conversion at fluid/solid boundary

$$\frac{c_{1L}}{sin(\theta_1)} = \frac{c_{2L}}{sin(\theta_{2L})} = \frac{c_{2S}}{sin(\theta_{2S})} \tag{4.1}$$

Where c_{1L} and c_{2L} are the longitudinal velocities in the fluid and solid respectively and c_{2S} is the shear velocity in the solid. The angles are taken from Figure 4.3. As the angle θ_1 is increased, two critical angles are encountered. The first critical angle is observed when θ_{2L} reaches 90^o and the longitudinal energy is either reflected or converted into an interface wave. At angles greater than the first critical angle only shear waves propagate in the second material. The second critical angle is observed at $\theta_{2S} = 90^o$ after which point no significant energy is propagated into the second material. All the energy is either reflected or converted into an interface wave. The two critical angles can be calculated by equations 4.2 and 4.3.

$$\theta_{2Lcritical} = sin^{-1}\left(\frac{c_1}{c_{2L}}\right) \tag{4.2}$$

$$\theta_{2Scritical} = sin^{-1}\left(\frac{c_{1L}}{c_{2S}}\right) \tag{4.3}$$

Surface waves are generated beyond the second critical angle and are marked by a decrease in the reflected wave [9]. The mechanism by which surface waves are produced is discussed in Section 4.4.

The next section briefly introduces the use of ultrasound for NDE.

4.3 Ultrasonic Testing

Ultrasonic **T**esting (**UT**) is a form of NDE that interrogates a sample by monitoring its response to high frequency sound [91]. UT can be used for flaw detection, dimension measurements or material characterisation. Figure 4.4 shows a typical longitudinal wave experiment, which can be configured with either 'pitch-catch' or 'pulse-echo' transducers. The transducer acts as an convertor between electrical and mechanical vibrations. In both cases the transducer is excited with an electrical signal that generates a longitudinal ultrasonic wave in the specimen. The receiver then converts the measured mechanical vibrations back into electrical signals for interpretation. In the pulse-echo configuration, the excitation transducer is also used to listen for reflections. In the pitch-catch configuration a second transducer is used to measure propagation through the specimen.

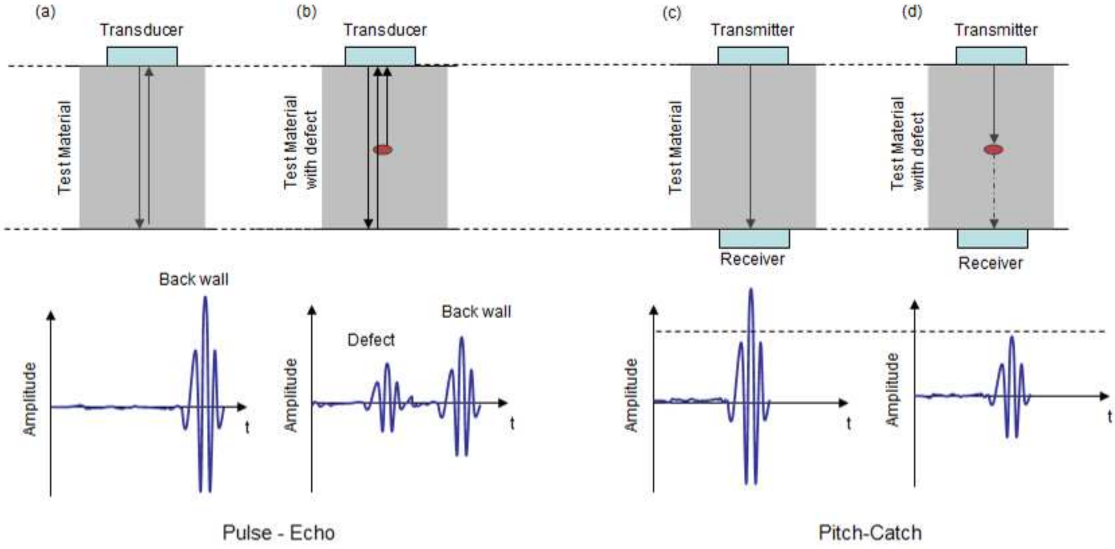


Figure 4.4: Ultrasonic testing schematics, showing pitch-catch and pulse-echo configurations.

If the propagation velocity is known, pulse-echo inspection can measure the specimen's thickness and the depth of any defects. A short ultrasonic pulse is generated which travels through the thickness of the material and is reflected by the back wall or any discontinuity, this is shown in Figure 4.4 a. The defect in Figure 4.4 b, generates an additional echo. The arrival time of each echo, can be converted to a distance using this simple equation:

$$distance = velocity \times time/2; \tag{4.4}$$

If the defect fails to reflect the wave back to the receiver, no echo will be observed. However, caution should be taken to ensure that this is not interpreted to mean that the sample is defect free. Defects may also be detected by monitoring the amplitude of the back wall reflection. A strong back wall signal provides validation that a defect is not present.

Pulse-echo measurements are widely used for thickness measurement where thickness accuracy as low as ±1 micron has be achieved by devices like the Olympus 25MX Plus [92]. Devices are currently available in hand-held configurations for on-site testing.

The pitch-catch configuration shown in Figure 4.4 c, measures the propagation of ultrasonic energy between transmitter and receiver. As shown in Figure 4.4 d, defects disrupt the signal, typically leading to a loss in amplitude and a phase shift. Additionally, if the distance between pitch-catch transducers is known, the experiment can be used to measure propagation velocity. This can be used for material characterisation.

It should be noted however, in both pulse-echo and pitch-catch configurations there are orientations of defects, such as a crack parallel to the direction of wave propagation, that are either difficult to detect, or appear artificially small.

4.4 Surface Wave Generation from Angled Wedge

The pitch-catch concept that has been discussed in the previous section can be extended to Lamb waves; this is detailed in Figure 4.5. The schematic shows two transducers positioned at a specific incident angle over a plate specimen. The plate is surrounded by air. The transmitter produces a longitudinal excitation which is converted to a Lamb wave through mode conversion. This was achieved by setting the incident angle so that the crests of the longitudinal excitation coincide with the crests of the desired Lamb wave resulting in constructive interference. This concept is known as the coincidence principle. The angle can be calculated using Snell's law [9], taking the longitudinal velocity in air as 340 ms^{-1} and the phase velocity of the Lamb wave from the dispersion curve in Figure 4.2. For example, to generate a 600kHz A_0 Lamb wave in 1mm thick aluminium, with phase velocity of approximately 2000 ms^{-1} (Figure 4.2), the transducers must be set at 9.8^o relative to the surface of the plate. The Lamb wave propagates forward, from A to B.

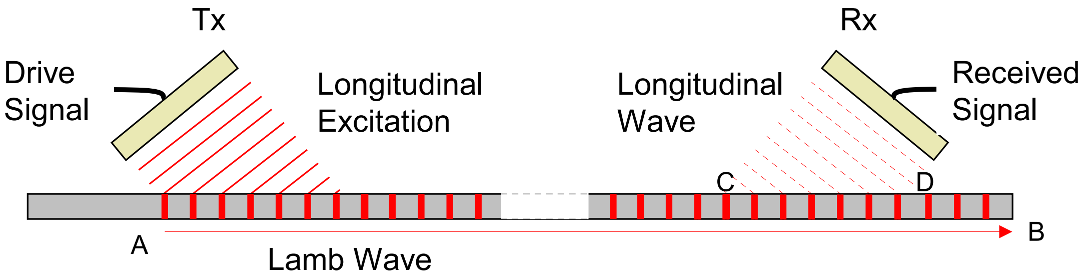


Figure 4.5: Schematic for Lamb wave generation and reception which shows two transducers at the appropriate incident angle positioned over a sample plate.

As the Lamb wave propagates, some of the energy 'leaks' out from the plate, generating a longitudinal wave in the surrounding medium. This wave is emitted from both the top and bottom of the plate, along the whole length of the plate. This wave can be measured using an ultrasonic transducer, placed at the appropriate incident angle so that the longitudinal wave reconstructs constructively on the transducer's face. Due to the symmetry of the arrangement, this angle is the same as the angle used in transmission. Figure 4.5, shows the receiver positioned at the optimal angle, measuring the longitudinal wave generated from the Lamb wave as it passed between points C and D.

The incidence angle is specific to the material properties and its 'frequency-thickness product'. For a fixed frequency configuration, a variable angle mounting is required to test different sample thicknesses. The angle be must accurately controlled since a 0.66^o misalignment can lead to a 50% reduction in the signal [93]. Angled transducers are most effective at generating the antisymmetrical mode A_0, which can have an out-of-plane displacement twenty times larger than S_0 [93] at typical working frequency thickness products (600Khzmm in this case). This results in superior sensitivity.

Using the same general principles as the aforementioned longitudinal wave UT, Lamb waves can be used to test the propagation path, or to listen for reflections. This will be discussed further in Chapter 6.

4.4.1 Inducing Ultrasonic Waves

It is useful to introduce the theory of ultrasonic propagation across material boundaries, before considering ultrasonic generation techniques.

4.4.1.1 Reflection and Transmission

When an ultrasonic wave hits the boundary between materials, some of the energy is transmitted through the boundary and the remainder is reflected back (assuming that the system is lossless) [94]. Consider the case of plane waves at normal incidence to the boundary. The amplitude of the transmitted pressure wave across a boundary can be calculated from the ratio of acoustic impedance:

$$T_{transmission} = \frac{2Z_2}{Z_1 + Z_2} \tag{4.5}$$

Where the specific acoustic impedance of a plane wave propagating in an isotropic lossless medium is defined as:

$$Z = \rho c \tag{4.6}$$

ρ= density of the material
c = speed of sound in the material

Acoustic impedance is typically specified in Rayls which are equivalent to newton-seconds per cubic meter (Nsm^{-3}).

Large mismatches in acoustic impedance at a boundary result in poor propagation of the ultrasonic wave. For maximum energy transfer, and hence SNR, it is preferable to couple the ultrasonic transducer to the specimen with a material of intermediate acoustic impedance. If the transducers are required to scan the sample, water is typically used for coupling, either by submerging the sample in a tank, or by spraying a water jet into the region between transducer and sample. Water coupling is not practical for the miniature robotic scanning system, which must rely on air coupling. The adverse effect of switching from water to air coupling will now be examined.

Consider the pitch-catch inspection experiment shown in Figure 4.6. The transducers are piezoelectric, and therefore convert between electrical and mechanical vibrations using the piezoelectric effect. The test specimen is rolled aluminium. The sound must travel from the transducer into the coupling fluid, then into the specimen, then back into the coupling fluid and finally back into the receiver transducer. Typical values for the acoustic impedance of each material are shown in Table 4.1, these have been taken from Ref. [95]. The 'through transmission' insertion loss can be calculated by multiplying the insertion loss at each boundary, this is shown in Table 4.2.

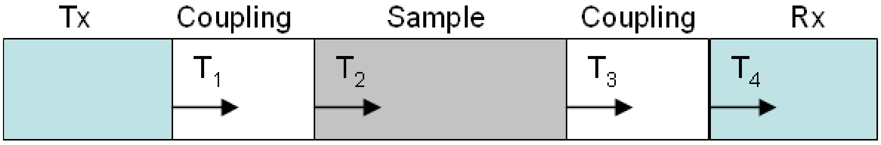


Figure 4.6: Example pitch-catch acoustic boundaries

Material Acoustic	Impedance (MRayls)
Piezoelectric ceramic:	33
Aluminium (rolled) :	17.33
Water:	1.5
Air:	0.000434

Table 4.1: Material Acoustic Impedance

Pressure Transmission Coefficient	Water-Coupled	Air-Coupled
T1	0.0870	2.63×10^{-5}
T2	1.8410	1.999
T3	0.1593	5.009×10^{-5}
T4	1.9130	1.999
Total	0.0487	5.296×10^{-9}
Insertion Loss	-26.23dB	-165.56dB

Table 4.2: Insertion loss calculations for each material boundary defined in Figure 4.6

Switching from water to air coupling increases the insertion loss by 139.33dB. The large reflection at the transmitter/air boundary also causes problems for pulse-echo systems as this very large initial reflection that can saturate the receiver amplifier. A method of reducing this step change in impedance is described in Section 4.5.3.

4.4.1.2 The Air Channel

The loss calculation in the previous section does not take into account any attenuation in the coupling medium or test sample. In solids and liquids this is very small, but in air it can be significant. The attenuation in the air channel is frequency dependent. Farlow [96] considered an air channel as a low-pass filter with a 3dB cut-off of 1.3MHz.

4.5 Non-Contact Ultrasonic Transducer Technologies

Creating and measuring ultrasonic signals requires conversion between electrical and mechanical vibrations; this is the function of the transducer. There are several established transducers available based on optical, electromagnetic, capacitive and piezoelectric techniques.

4.5.1 Laser Generated Ultrasound

When a surface is illuminated by a laser, a number of different physical process may occur [97]. At low powers these include heating, the generation of elastic waves and in materials such as semiconductors, the generation of electric currents. At high powers, the material on the surface can removed and a plasma formed, the material in the sample may melt, suffer plastic deformation and even crack. Only low power excitation is suitable for inspection, since they do not significantly damage the sample.

When electromagnetic radiation from a low power laser is absorbed by the surface region of the sample, it causes heating. Thermal energy propagates into the specimen as thermal waves. This causes the heated region to expand, causing thermoelastic stresses, which in turn generate elastic (ultrasonic) waves. These waves propagate through the sample. Laser generated ultrasound typically has a wide bandwidth (tens of MHz) and fine spatial resolution. Lasers can launch bulk, surface and guided waves [98, 99].

Laser interferometers measure the varying phase difference between a reflected and reference laser beam to accurately measure small displacements. They can be used to measure the surface component of ultrasonic waves.

However, for transmission, lasers typically require too much power to make them suitable for use on the miniature robotic vehicles. In addition to this they tend to be too large for practical use on the vehicles.

4.5.2 Electromagnetic Acoustic Transducers

Electro**M**agnetic **A**coustic **T**ransducers (**EMAT**s) harness the electromagnetic acoustic effect to induce and measure ultrasonic waves [100, 101, 98]. An EMAT transducer is created by placing a coil on the surface of the sample, then placing a permanent magnet behind the coil. The coil is excited with a high frequency signal resulting in an oscillating magnetic field. This changing magnetic field generates an electromotive force, which produces eddy currents on the surface of the sample. The static magnetic field generated by the permanent magnet interacts with the eddy currents. The resulting Lorentz force causes the electrons to oscillate at the excitation frequency. The whole lattice follows this excitation, causing a mechanical vibration at the excitation frequency. This propagates as an ultrasonic wave. The reciprocal of this effect allows EMATs to act as receivers as well.

EMATs are non-contact and can excite and detect a large variety of modes depending on the coil design. They are robust and can work at high temperature, but generally have low sensitivity and require the test material to be conductive. EMAT transmitters require a high drive current making it difficult to embed them into the RSAs. EMATs receivers show some potential for use on the vehicles as omnidirectional sources, but have not been investigated in this work.

4.5.3 Piezoelectric Transducers

Piezoelectric transducers are based on the well documented piezoelectric effect [98]. Applying an electrical signal generates a mechanical vibration which can be used to induce acoustic waves. The reciprocal effect makes piezoelectric ceramic transducers suitable for reception. Piezoelectric transducers are typically used at resonance, with the bandwidth dependant on the amount of damping. There is very little damping in air-coupled transducers, making such systems narrow band. The thickness mode resonance is dependent on the thickness of the transducer; reducing the thickness increases the resonance frequency.

Conventional piezoelectric transducers have high acoustic impedance ($\simeq 33$ MRayls), giving poor acoustic matching with low density loads such water or air. They also suffer from performance loss due to lateral oscillation in the crystal. Piezoelectric composite transducers have been developed to address these issues. In the most commonly used 1-3 structure, the ceramic is diced into square based pillars that are surrounded by a passive polymer matrix. This is shown in Figure 4.7. The polymer reduces the transducer's average acoustic impedance and breaks up lateral oscillation.

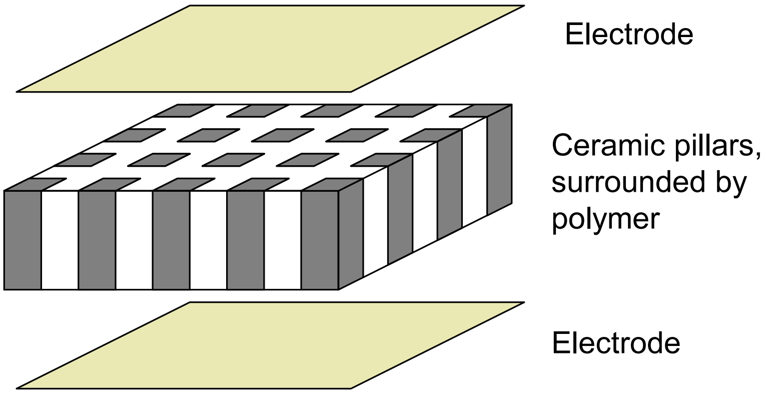


Figure 4.7: Exploded view of 1-3 composite piezoelectric transducer

The Centre for Ultrasonic Engineering (CUE) has conducted extensive research into the optimum piezoelectric composite transducer design for use in air [102]. It was found that for transmission, maximum performance occurs at volume fractions greater than 70%. (Volume fraction is defined as the volume of active ceramic relative to the total volume of the transducer. The remaining volume is made of the passive polymer.) In reception, low volume fractions in the range of 10% - 15% offer optimal sensitivity, however volume fractions in the order of 20% are required for reliable manufacture. (The manufacturing process firstly dices the piezoelectric ceramic to create the columns and then fills the voids with polymer. If the volume fraction is less than 20% the ceramic tends to disintegrate when diced.)

R. Banks has demonstrated piezoelectric transducers with increased bandwidth, using novel mechanical design [103]. A standard 1-3 piezoelectric composite transducer was machined into a cone shape so that the resonance frequency increased towards the outer edge of the transducer. Banks demonstrated a piezoelectric composite transducer with 300 kHz bandwidth and a SNR of 18 dB.

4.5.3.1 Matching Layers

Applying a mechanical matching layer to the front face of a composite transducer minimises the step change in acoustic impedance, increasing energy transfer. There is currently no material available with the acoustic properties that would provide ideal quarter wavelength matching between a piezoelectric transducer and air. However, a novel approach by Kelly et al.[104] using porous paper partially submerged in silicon rubber provides a 30dB reduction in insertion loss for an air-coupled pitch-catch system.

Wheel probes [105] are an alternative to fluid coupled ultrasound. They consist of a transducer immersed in liquid inside a hollow wheel. They provide non-bonded solid coupling, which obviates the signal attenuation problems of air coupling, but are mechanically much more complex and suffer from significant performance degradation if the surface is rough or unclean.

4.5.4 Electrostatic Transducers

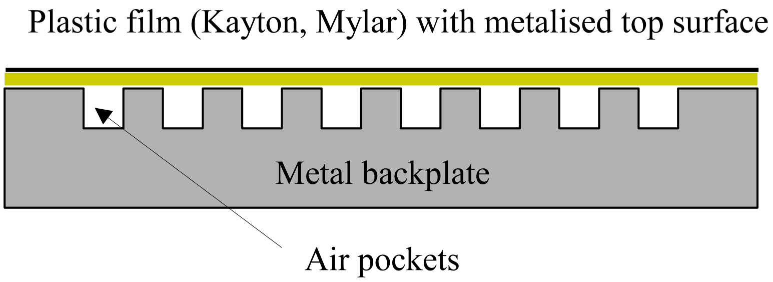


Figure 4.8: Schematic diagram for electrostatic transducer

Electrostatic transducers are formed by a thin dielectric film with a conductive film coated on the exterior front surface, stretched across a conducting backplate. This is shown in Figure 4.8. The membrane sandwiches the dielectric and forms a capacitor. The conductive backplate and conductive top layer form the two electrodes. A bias voltage is applied to the membrane which becomes electrostatically attracted to the backplate, pulling it down, thus trapping small pockets of air. When an additional time varying voltage is applied to the electrodes, the electrostatic force oscillates causing the membrane to vibrate over the air pockets. This generates an ultrasonic wave. The frequency of vibration is related to the size of the air pockets, with larger air pockets producing lower frequency vibrations.

Reciprocally, if a pressure wave is applied to the surface, an electrical signal is generated that can be measured on the electrodes. Electrostatic transducers typically have a wide bandwidth and are intrinsically well coupled to air. However, the fine film on the front of the transducer makes them fragile, rendering them unsuitable for most industrial environments. Electrostatic transducers typically have 20dB lower sensitivity than well matched piezoelectric composite transducers [95], making them less attractive for use in the ultrasonic payload at this point.

This concludes the brief overview of ultrasonic testing. The chapter will now focus on the implementation of the embedded air-coupled ultrasonic payload.

4.6 Overview of Embedded Non-Contact Ultrasonic Inspection Payload

The embedded non-contact ultrasonic inspection payload is capable of exciting and measuring 600kHz A_0 Lamb Waves in thin steel or aluminium plates. As discussed in Section 4.4, the A_0 mode generates significantly more out-of-plane displacement which makes it suitable for generation from angled transducers. This operating frequency was chosen as a trade off between temporal/spatial resolution and signal attenuation, hence SNR of the system. Piezoelectric composite transducers were selected for their excellent sensitivity.

The payload electronics consist of three components; the payload controller (including ADC), the transducer front end and the receiver preamplifier. These are illustrated in Figure 4.9 and discussed in Sections 4.8, 4.9 and 4.10 respectively. Section 4.11 details system integration. Finally Section 4.12 evaluates the payload's performance.

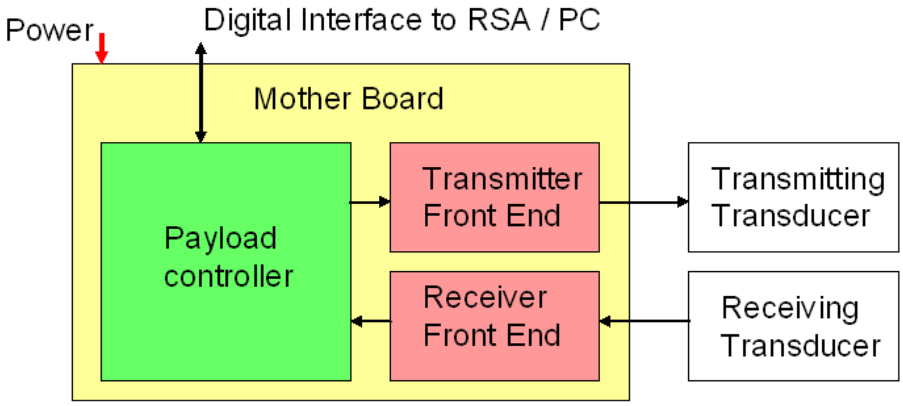


Figure 4.9: Block diagram of ultrasonics payload

4.7 Payload Transducers

The payload uses a pair of 30mm square 1-3 piezocomposite transducers designed to operate at the desired 600kHz, in pitch-catch mode. Following the transducer design guidelines of Kelly [95], the transmitting transducer had a 70% volume fraction of PZT-5H and the receiving transducer had a 30% volume fraction of PZT-5A. In both cases, the passive filler material was CY1301/HY1300. A matching layer was integrated onto the front-face of each transducer to minimise insertion loss due to the impedance miss-match between the transducer face and air. The matching layer was created by curing a thin layer of silicon rubber onto membrane filter paper which gradually reduces the acoustic impedance as the energy passes from the silicon through to the porous paper and into the air.

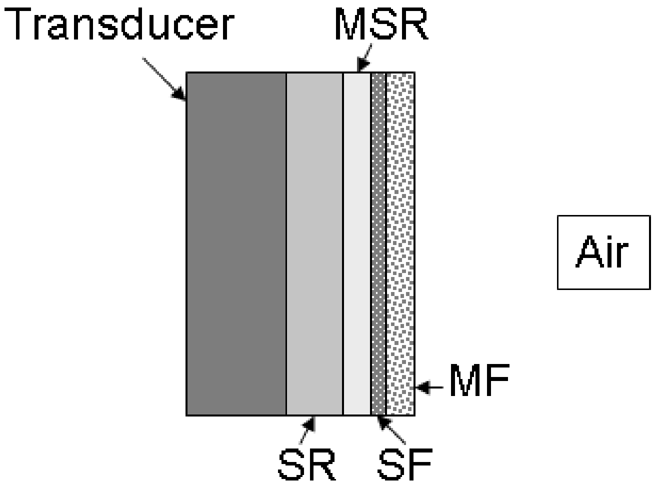


Figure 4.10: Illustration of multilayer matching to air

Kelly et al [104] used an electron microscope to examine this design and modelled the matching layer as four sub layers, these are shown in Figure 4.10. The transducer interfaces with a layer of silicon rubber (SR). A portion of the silicon rubber is modified by the presence of the membrane filter (MF) and is termed modified silicon rubber (MSR). The third layer is a result of the membrane filter absorbing silicon rubber and is termed 'saturated filter' (SF). The thickness of the MSR, SF and MF were found to be constant and these thickness are documented in Table 4.3. The thickness of the silicon rubber is optimised for minimal insertion loss and following the advice of Kelly et al, was set to 0.2mm for transmission and 0.64mm for reception.

The material properties of each layer is shown in Table 4.3.

Layer	Thickness (mm)	Density (kgm^{-3})	Atten 1 MHz ($dBcm^{-1}$)	Elastic Stiffness (Nm^{-3})
1. SR	Variable	951	10	9.41e8
2. MSR	0.08	951	18	3.09e8
3. SF	0.01	697	100	6.97e8
4. MF	0.11	360	120	4.2e7

Table 4.3: Characteristics of the matching layer's sub-layers.

By integrating this matching layer into both the transmitting and receiving transducers, the transmission pressure magnitude is increased by a factor of 8.7 and the overall system insertion loss is reduced by 30dB [104].

4.8 Payload Digital Circuitry

The payload is capable of recording the full time domain signal from the the receiving transducer. A typical piezoelectric composite receiver signal is narrow band, with a centre frequency of 0.5 - 1MHz.

The payload uses a dsPIC microcontroller from Microchip [106], which is a 16bit family of ICs with DSP functionality and relatively fast ADC's. The microcontroller's do not have a communications port fast enough to stream the ultrasonic signal in real time, so the signal must be buffered on-chip. The dsPIC33f706 [107] features a 1.1MHz 10bit ADC and 16kBytes RAM. This can potentially store up to 7ms of time domain data in RAM. The A_0 Lamb wave at 600kHz travels at approximately 3mm/μs, so 7000μs corresponds to 21m of travel.

There is a common misconception that the Nyquist sampling criterion states that the sampling frequency must be twice the highest frequency component. This is only true if the signal contains frequencies right down to the DC. If the signal has limited bandwidth, the sampling frequency only needs to be twice the bandwidth and not twice that of the highest frequency.

The following section details how a signal with a frequency greater than the Nyquist frequency can be sampled as long as the bandwidth is less than the Nyquist frequency. This allows the use of an ADC with a Nyquist frequency of 550kHz to sample a 600kHz band-limited signal.

4.8.1 Sampling Bandwidth Considerations

Consider a 600kHz tone burst of 10 cycles. The payload uses narrow-band transducers, so in order to provide a realistic narrow-band receiver signal, the tone burst has been windowed by a Gaussian function to limit the frequency components. Figure 4.11 shows this tone burst, sampled at 1MHz and 20MHz. The signal sampled at 20MHz is accurately captured, but the tone burst sampled at 1MHz is aliased down to -400kHz and 400kHz.

The signal can be zero padded to reveal the lost frequencies. This is achieved by inserting zeros between samples, i.e. to zero-padded up to 20MHz, 19 equally spaced zeros were inserted between each sample. This is shown in Figure 4.12. Notice how the 600kHz peak has been revealed along with a peak at 1.4MHz. These peaks repeat up to the Nyquist frequency of 10MHz. In order to extract the frequency

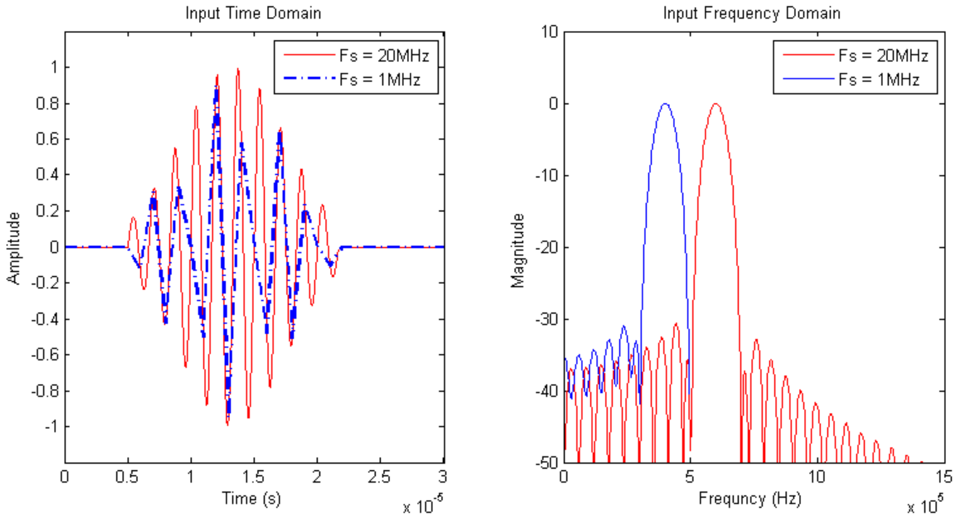


Figure 4.11: Example of under sampling a signal

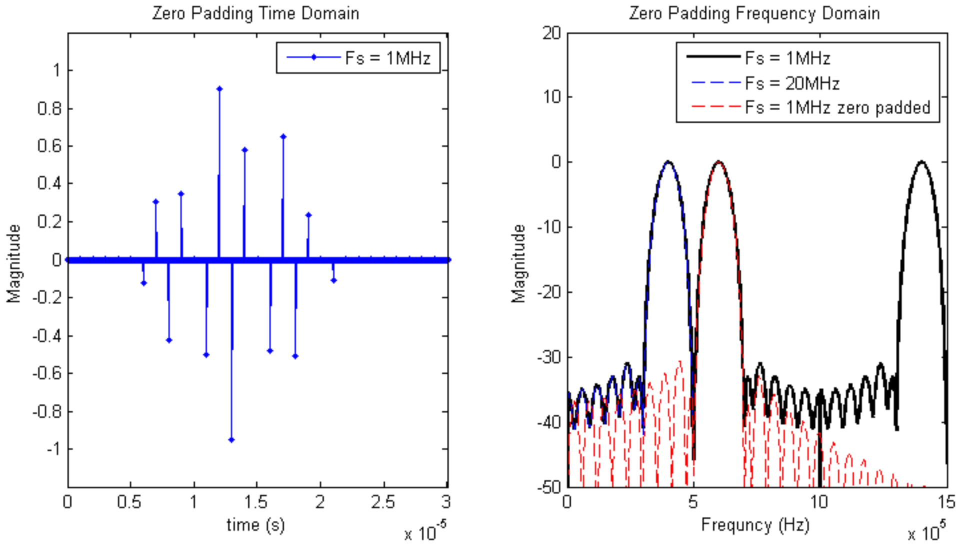


Figure 4.12: Example of zero padding an under sampled signal

of the tone burst, the signal must be band pass filtered to remove the peaks below 500kHz and the peaks above 1MHz. The result, which is shown in Figure 4.13, shows a good representation of the 600kHz tone burst.

This technique can be used to sample the 600kHz signal with the 1.1MHz ADC on the dsPIC33F. It should be noted that if the signal had contained information in the 400kHz region it would have been combined with the aliased 600KHz signal and there would not have been enough information to reconstruct the original waveform. However, since the transducers have limited bandwidth this can be avoided by selecting a sample frequency to match the transducer. The receiver electronics contain a band-pass filter to ensure that all frequencies outwith the sampling bandwidth are removed before sampling.

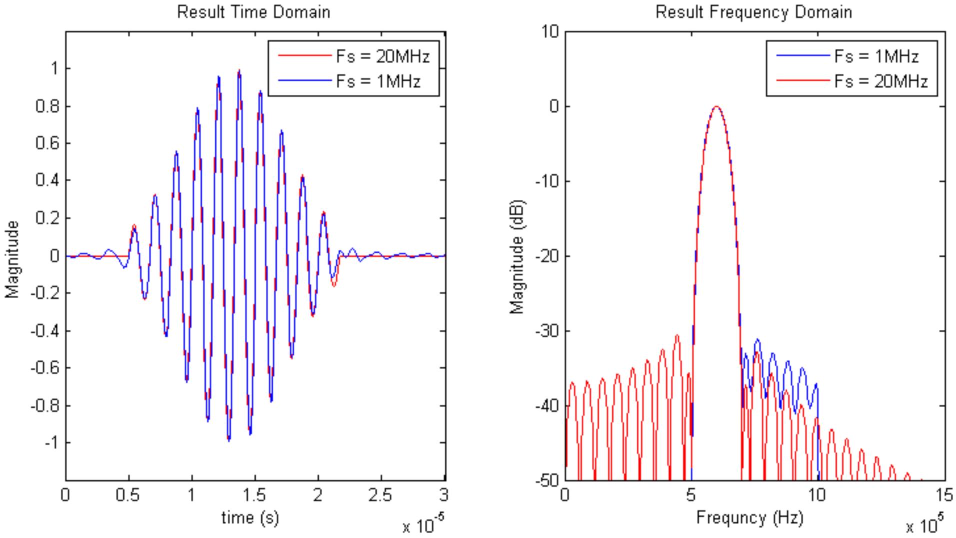


Figure 4.13: Example of band pass filtering a zero-padded signal

4.9 Payload Transmitter Front-End

The embedded electronics produce the excitation signal used to drive the ultrasonic transmitter. The electronics can generate a 100V unipolar square wave or a ±15V sine wave. The frequency and number of cycles in a burst frequency can be controlled in software. A block diagram for the system is shown in Figure 4.14.

The drive electronics are essentially two separate sub-systems, both driven by the host microcontroller. The square wave can be produced at a higher voltage with less circuitry, however the sine wave offers a more elegant excitation signal.

4.9.1 Generation of Square Wave Excitation Signal

The host microcontroller generates a TTL square wave burst using an GPIO pin controlled by a timer. This drives a BS108 MOSFET in drain-source configuration which switches a 100V DC voltage over the transducer. The schematic is shown in Figure 4.15. The BS108 was selected for its fast switching capabilities. When Q1 is off, the supply voltage is present across the transducer. Activating Q1 shorts the transducer to ground. The high voltage is created using a specialised DC-DC convertor, the Q01-12 [108] that can create 100V with a power rating of 1W. The circuit diagram is shown in Figure 4.15.

C1 acts as an energy reservoir so the DC-DC convertor does not need to supply the instantaneous peak current to the transducer. C1 is large enough that the energy reservoir is only slightly reduced during transducer excitations, it is then replenished between excitations.

Figure 4.16 details the operation of the square wave driver. When Q1 is active, the capacitor discharges through Q1 which has an on resistance R_{MOSFET}. The time constant can be calculated as τ_{FE}. The transducer has a capacitance C_{TX}.

$$\tau_{FE} = R_{MOSFET}C_{TX} \tag{4.7}$$

The minimum voltage (V_{min}) can be found by considering the ratio of Q1's on resistance (8Ω) and the load resistance R_{load}

$$V_{min} = \frac{R_{MOSFET}V_{Supply}}{R_{load}} \tag{4.8}$$

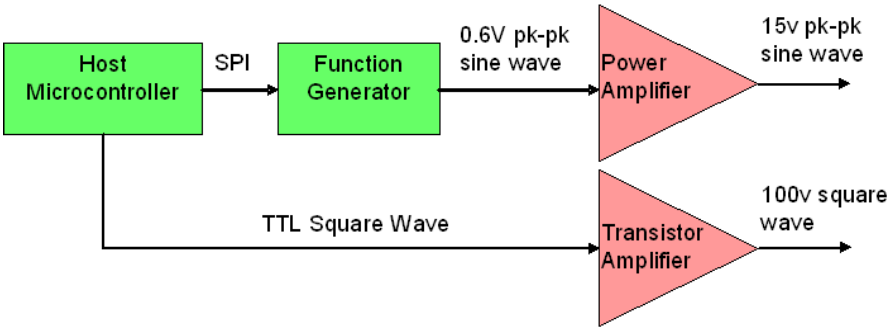


Figure 4.14: Embedded transmitter block diagram

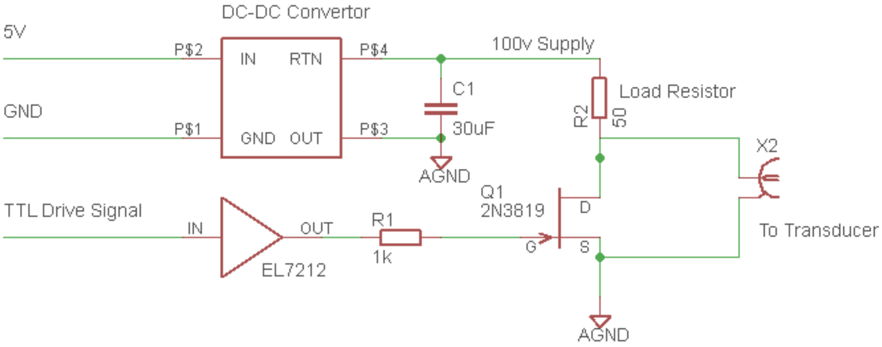


Figure 4.15: Simplified schematic of the square wave driver

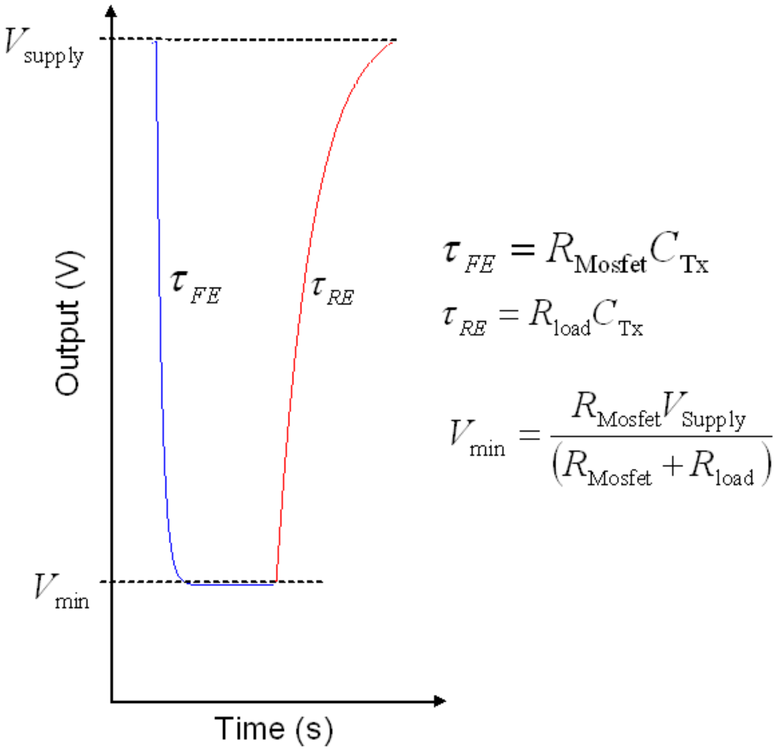


Figure 4.16: A time domain model of the square wave driver

When Q1 is active, the transducer charges along the time constant, τ_{RE}.

$$\tau_{RE} = R_{load}C_{TX} \tag{4.9}$$

Since the transducer must be switched at the drive frequency of approximately 600kHz, it must discharge and recharge in approximately 1.667us. This is not always possible if the transducer has significant capacitance.

Consider an example where the transducer has a capacitance, C_{TX}, of 10nF and the required switching frequency is 600kHz. The only variable that can be tuned is R_{load}. As R_{load} is reduced, τ_{RE} decreases, leading to faster switching, hence maximum voltage swing. However, reducing R_{load} increases V_{min} reducing the voltage swing. Figure 4.17 shows a plot of R_{load} against output voltage swing. The time trace shows the modelled excitation waveform using the optimum value of R_{load}. Figure 4.18 shows the experimentally measured output waveform for open circuit output (no transducer) and Figure 4.19 shows the experimentally measured output waveform when a 30 × 30 mm transducer was attached. Figure 4.19 exhibits a reduced voltage due to insufficient charging time between cycles. In addition to this the capacitive load causes significant ringing.

Future iterations of the circuit may replace the BS108 with another MOSFET with lower on resistance or use multiple MOSFETs in parallel to reduce R_{Mosfet}.

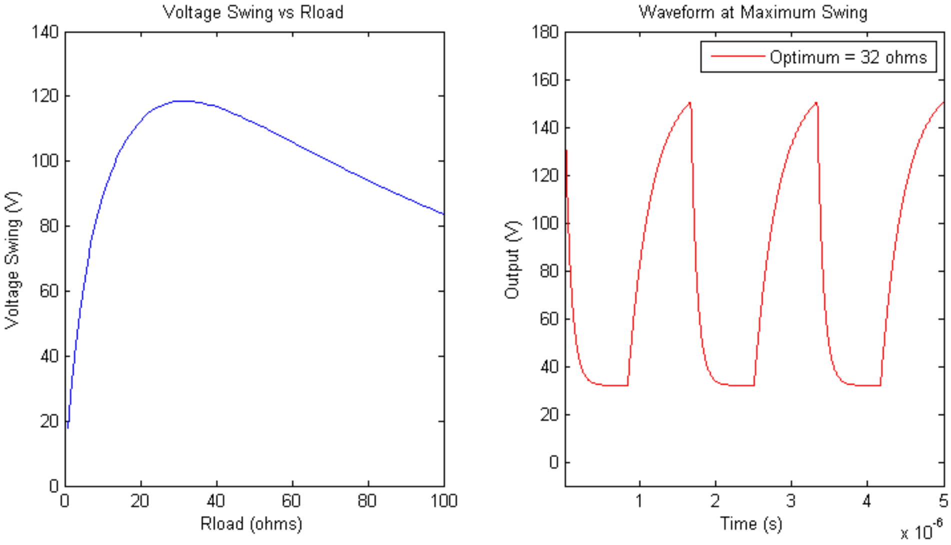


Figure 4.17: Optimum load resistor for square wave driver

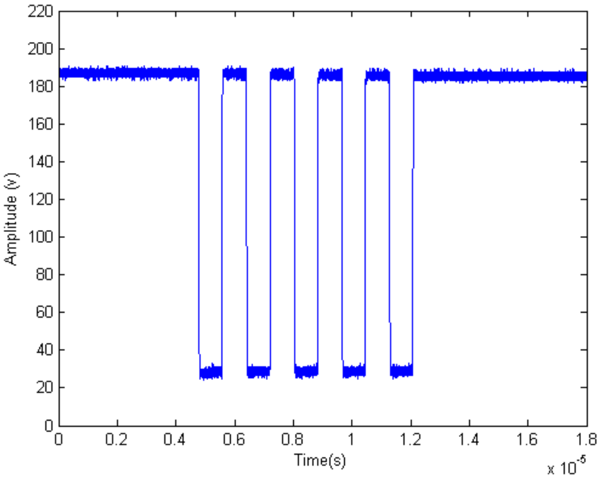


Figure 4.18: 600kHz square wave excitation, 5 cycles unloaded

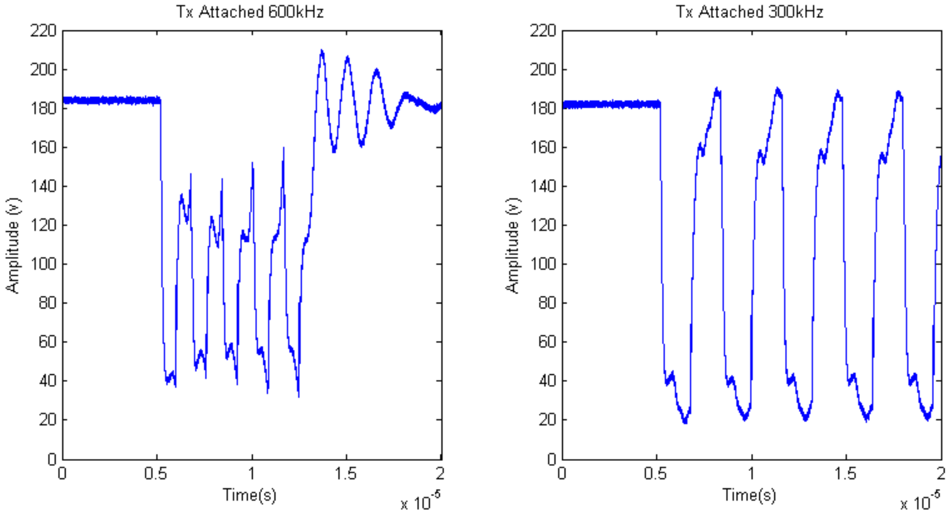


Figure 4.19: Amplified 600kHz and 300kHz excitation, 5 cycles, Tx attached

4.9.2 Generation of Sine Wave Excitation Signal

Embedded digital to analogue convertors (such as those found in microcontrollers) are typically unable to produce signals of frequency as high as 600kHz. The conventional approach to high frequency sine wave generation is to use a sine wave generator IC to create a constant sine wave and then to blank out the periods that are not in the burst. This approach requires significant analogue circuitry and makes it difficult to accurately control both the frequency and number of cycles, and ensure that the tone burst begins and ends at zero phase. Recent advancements in mixed analogue and digital integrated circuits provide a more elegant solution. The AD5930 [109] function generator can produce a sine or triangular signal at frequencies up to 25MHz. It can be programmed to produce continuous, burst or sweep outputs of the required characteristics. This device was interfaced with the host microcontroller using a serial interface (SPI). It can be configured in software to generate at tone burst of specific frequency and number of cycles.

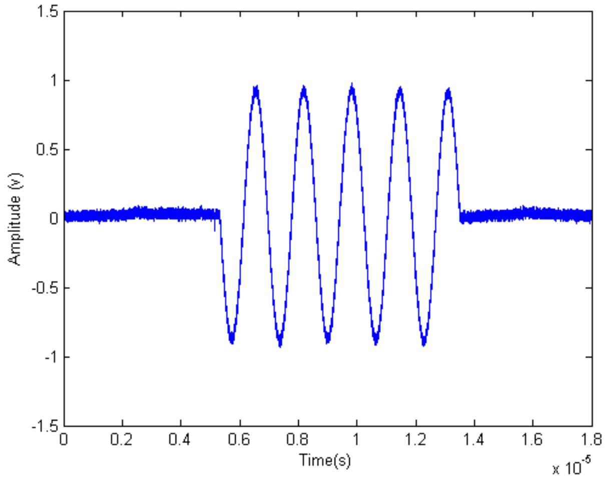
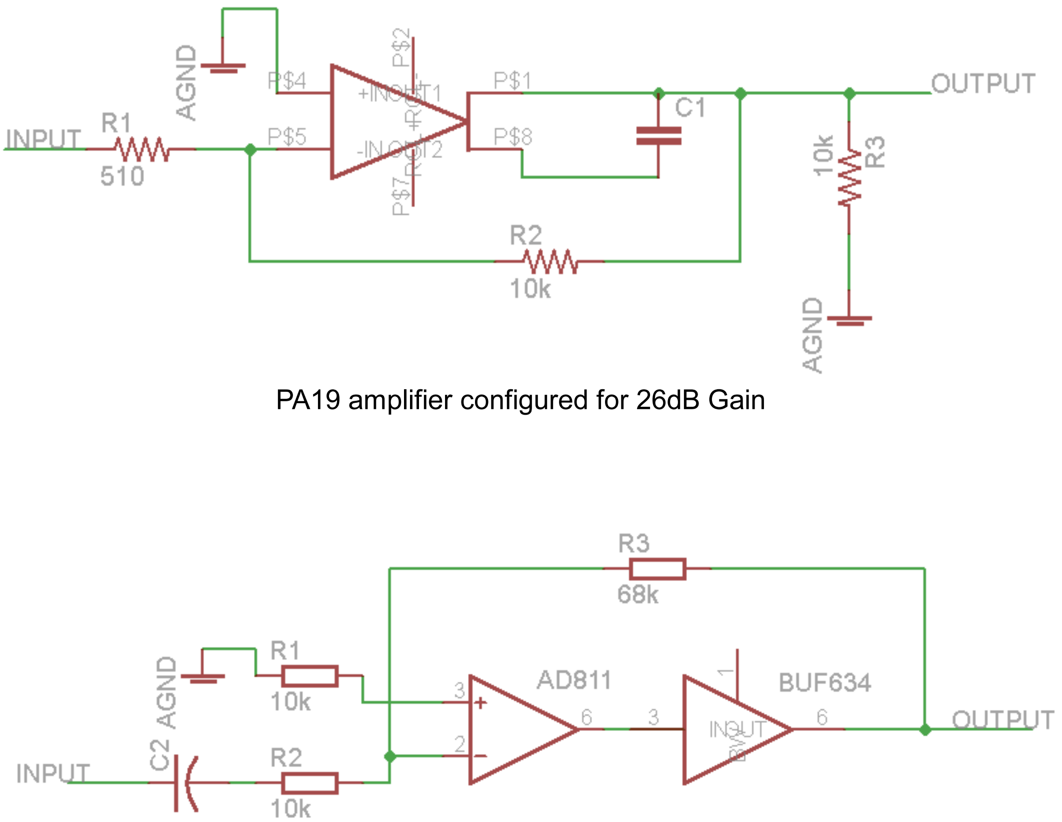


Figure 4.20: A 5 cycle 600kHz tone burst generated by the AD5930 sine wave generator

The output of the function generator is buffered and amplified to $2V_{pk-pk}$ by an AD8058 as shown in Figure 4.21. The signal is then amplified to the required excitation voltage. Two approaches to this high power amplification were evaluated. Firstly the PA19 [110] which is a high voltage ($\pm 40V$), high current (4A peak) operational amplifier. The PA19 was able to drive piezoelectric composite transducers at up to almost ($\pm 40V$), but was not ideal for miniaturisation due to the large footprint (30mm x 30mm) and high quiescent current of 100mA at $\pm 15V$ which limited battery life. An alternative approach used the AD811 [111] from Analogue

Devices with a Texas Instruments BUF634 [112] on the output, to supply the high drive current. This approach has a considerably smaller footprint (approximately 10mm x 20mm dependent on layout) and a lower quiescent current of approximately 10mA. The main drawback from this approach is that the drive voltage is limited at $\pm 15V$. The schematic for both circuits is shown in Figure 4.21. Figure 4.22 shows the experimentally measured output from the AD811 amplifier, both open circuit and when connected to a 30 × 30 mm 600kHz transducer. There was minimal degradation to the signal when the transducer was attached.



PA19 amplifier configured for 26dB Gain

Ad811 amplifier configured for 17dB Gain

Figure 4.21: Schematics for two power amplifier circuits. The first circuit can amplify to $\pm 40v$. The second can only amplify to $\pm 15V$, but has a smaller footprint and only draws 10mA quiescent current

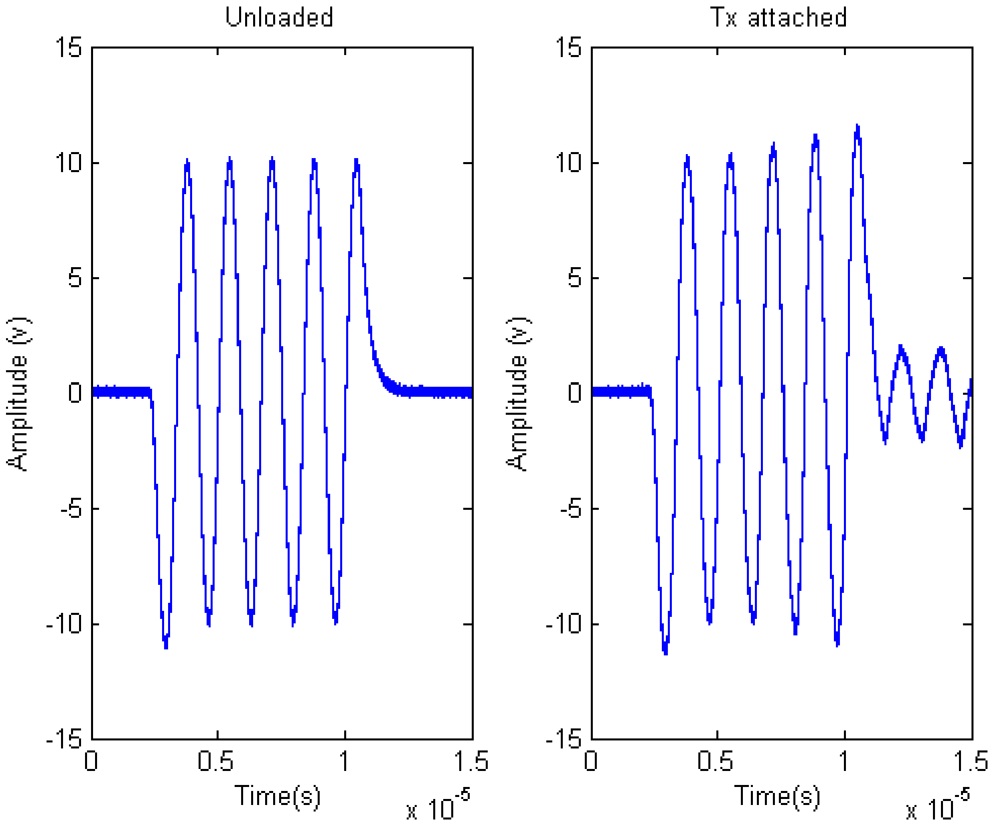


Figure 4.22: Amplified 600kHz 5 cycle tone burst excitation using AD811 amplifier. The graph on the left shows the open circuit output. The graph on the right shows the output, when attached to a 30 × 30 mm piezoelectric composite transducer.

4.9.3 Comparison of Sine and Square Wave Excitation

Using Fourier theory, a square wave can be decomposed into a sum of sine waves at odd harmonics of the fundamental frequency (1×, 3×, 5×, etc). This is formalised in Equation 4.10. An example is shown in Figure 4.23. A piezoelectric composite transducer used at resonance is narrow band. If the fundamental frequency of the square wave excitation is matched to the transducer resonance, the transducer will essentially react mainly to the fundamental frequency. Interestingly the amplitude of this fundamental component is $\frac{4}{\pi}$ larger than the square wave. If the same transducer is driven with a $10v_{pk-pk}$ sine wave and a $10v_{pk-pk}$ square wave, the output pressure will be 27% larger than when driven with a square wave. The square wave driver provides best results when using piezoelectric composite transducers, because it injects more energy into the system, leading to better SNR.

$$square(\omega, t) = \frac{4}{\pi} \sum_{x=0}^{\infty} \frac{1}{2x+1} sin((2x+1)\omega t) \tag{4.10}$$

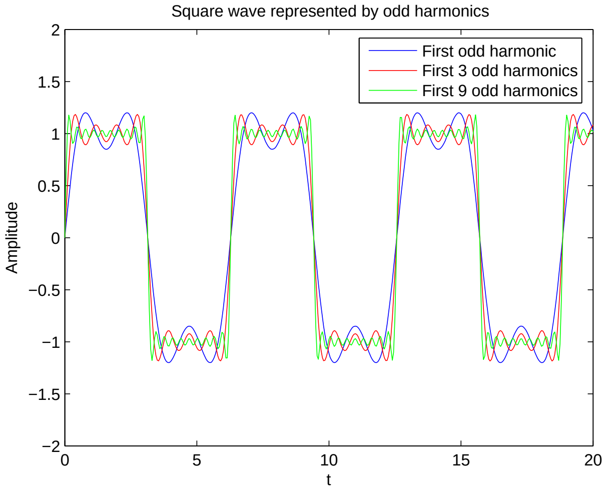


Figure 4.23: This Figure shows how a square wave can be represented as a sum of sine waves at odd harmonics of the fundamental frequency. The approximation improves with the number of components used

The sine wave driver provides a more elegant drive signal which may be more suitable for wide band transducers such as electrostatics that would react to the harmonic frequencies filtered out by the narrow-band piezoelectric composite transducers.

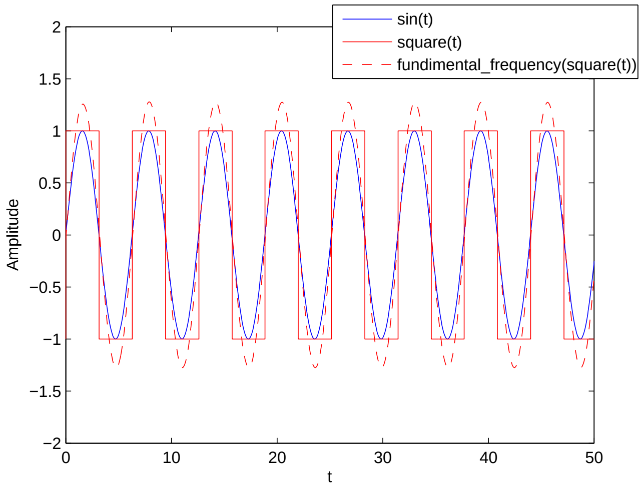


Figure 4.24: This Figure shows a sine and square wave at the same amplitude and frequency. The dashed line shows the fundamental frequency component of the square wave, note that it is $\frac{4}{\pi}$ larger than the sine wave.

4.10 Payload Receiver Front-End

The receiver schematic is shown in Figure A.5. The receiver front end uses an AD8058 [113], configured as an inverting voltage amplifier to provide the initial gain stage. 'R3' is the circuit's input impedance and is kept high for maximum voltage transfer. The ratio of 'R4' and 'R3' set the initial gain. The operational amplifier is configured for single supply, with 'C7' and 'C3' decoupling the DC offset. A common-source junction **F**ield-**E**ffect **T**ransistor (**FET**) amplifier is used with an LC tuner to provide additional gain (60dB) and band pass filters the signal. The centre frequency of the filter is the resonant frequency of the LC oscillator, this can be calculated as follows:

$$f_c = \frac{1}{2\pi\sqrt{L_1, C_6}} \tag{4.11}$$

Where 'L1' and 'C6' are defined in Figure A.5. Diodes D1 - D4 clip the output to $\pm 1.3V$ preventing large inputs from damaging subsequent circuitry (particularly after amplification). The experimentally measured frequency response of the complete circuit is shown in Figure 4.25. The value of 'R4' is tuned so that receiver signals fill a good proportion of the $\pm 1.3V$ voltage range visible to the ADC, minimising the quantisation error.

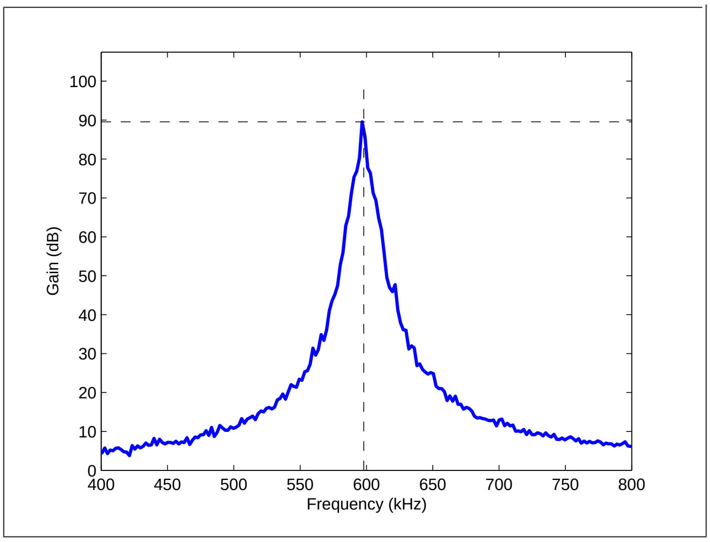


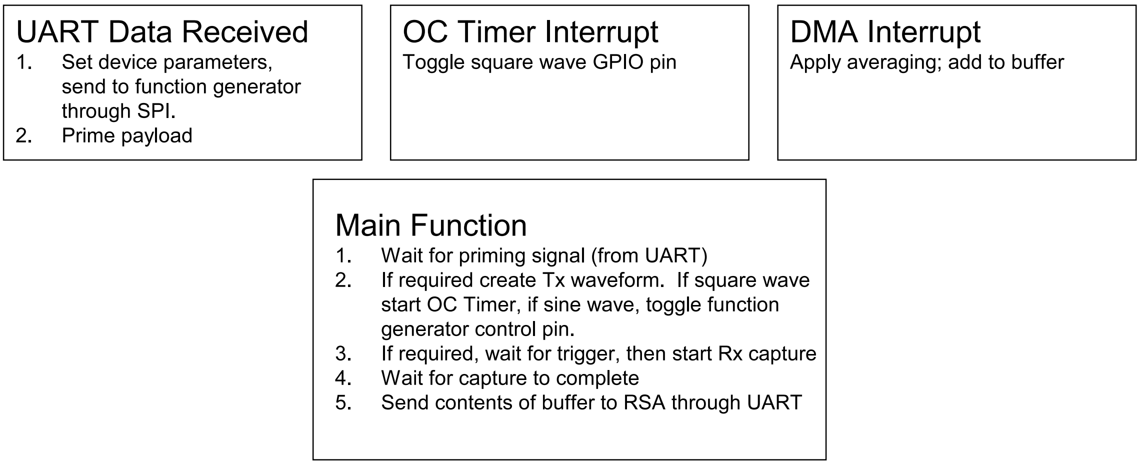
Figure 4.25: Frequency response for embedded preamplifier

The peak output from the receiving transducer is in the order of hundreds of nano volts for pitch-catch air-coupled Lamb wave inspection. Minimising interference with this very low amplitude signal is critical for performance. Configuring the input as a differential amplifier for common mode noise rejection has the potential to reduce interference and will be considered in future work. In addition to this, it is important to maximise the electrical shielding of the amplifier. Good shielding was achieved in this project by embedding the preamplifier into the receiver's enclosure and shielding the entire assembly. This is discussed in Section 4.11.

The noise performance of the system is heavily influenced by the performance of the preamplifier, which has to amplify signals with extremely low amplitude. Operational amplifiers are designed with low source impedance on the inputs, so only voltage noise is significant, the current noise, including Schottky noise, is swamped by the input impedance [114]. At room temperature, ultra low noise devices can operate with input voltage noise as low as $0.8nV/\sqrt{Hz}$, this has been achieved by devices such as the OPA847 [115]. The AD8058 that has been used in this work has an input voltage noise of 7 $nV/\sqrt{Hz}$, making it suboptimal. Future work will replace the AD8058 with an ultra low noise voltage feedback amplifier such as the OPA847.

4.11 System Integration

4.11.1 Microcontroller Software



UART Data Received

1. Set device parameters, send to function generator through SPI.
2. Prime payload

OC Timer Interrupt

Toggle square wave GPIO pin

DMA Interrupt

Apply averaging; add to buffer

Main Function

1. Wait for priming signal (from UART)
2. If required create Tx waveform. If square wave start OC Timer, if sine wave, toggle function generator control pin.
3. If required, wait for trigger, then start Rx capture
4. Wait for capture to complete
5. Send contents of buffer to RSA through UART

Figure 4.26: Overview of microcontroller code

The structure of the host microcontroller's firmware is shown in Figure 4.26. The payload and robot processor communicate through a serial interface which is managed by a UART on the payload side. The RSA sends a ten byte packet to the payload containing details of the frequency, number of cycles, output type and number of averages. The payload is configured as required, sending configuration information to the function generator through SPI. The circuit generates the excitation waveform either by using an output compare (OC) timer to control a general purpose pin (GPIO) to create a square wave, or by triggering the function generator to create a sine wave. Once the excitation is complete, the ADC is activated. Direct Memory Access (DMA) gives the ADC direct access to the RAM (bypassing the CPU) which is required for the 1.1MHz ADC operation. The DMA interrupt is activated every 256 samples and the data is then passed to a reception buffer. There is the facility to implement averaging by only adding a fraction of the new signal to the buffer and repeating the entire capture several times. Given that the ADC packet must be processed in the short time window before the next sample arrives, only averaging by a factor of two is possible (it can be simplified to the less computationally expensive action of shifting bits rather than a full division). The ADC is deactivated when the receiver buffer is full (typically set at approximately 2000 samples). This whole process is repeated for the total number of averages, before the result is sent through a UART to the robot.

4.11.2 Electrical Interference and Noise

The intrinsic nature of a design featuring a high voltage driver in the same system as a sensitive preamplifier lead to noise on the receiver signal. The preamplifier discussed in Section 4.10 was extremely susceptible to noise on the supply lines. For this reason, the preamplifier was powered from a separate 12V battery (MN21) regulated to 10V by a low drop-out linear regulator. This solution provided optimal SNR at the cost of increased size and the slight inconvenience of having two batteries on the robot. The MN21 battery provided 3 hours of operation.

The high voltage DC-DC convertor used to supply the square wave driver (Section 4.9.1) required a 12V supply. This was connected directly to the battery. There is a slight drop in the high voltage line as the battery drains and the 12V supply approaches 10V. However, this is inconsequential, since the voltage with a 10V input is still greater than the maximum voltage swing possible when driving a typical 600kHz transducer with the square wave driver circuitry (See Section 4.9.1 for details).

4.11.3 Electronics Configuration

The AD5930 function generator and dsPIC33F microcontroller were packaged onto a single board with an AD8058 to buffer the output of the function generator and input to the ADC. The required input/output lines were routed out to a 40 pin surface mount 1mm pitch connector. The schematic is shown in Figure A.3, Appendix A.

The individual transmit and receive circuits were packaged into the transducer enclosures. This minimised the electromagnetic interference emitted from the high voltage transmitter and maximised the noise immunity of the receiver. The digital circuit board slotted into an adapter board that formed the third layer of the RSA PCB stack (See Section 2.7, Chapter 2). The adapter board contains miniature BNC connectors to attach the transducers. This modular design has several advantages. Firstly, the payload can be configured to act as a transmitter, receiver or transceiver depending on the specific requirements. Modules can be swapped depending on the transducer technology used, for example if using electrostatics, the front-end must provide a high voltage bias that is not required for other transducer technologies. Finally, specific modules can be upgraded without changing the overall design.

4.11.4 Acoustic Interference

In order to perform localised pitch-catch experiments, a pair of ultrasonic transducers must be mounted onto the RSA in close proximity. This provides a direct mechanical path between transmitter and receiver leading to a multi-path acoustic problem. The problem is shown graphically in Figure 4.27.

The large acoustic miss-match between the transducer/sample and air make the received signals very small (See Section 4.4.1.1), compared to the direct ultrasonic path through the chassis. The energy coupling directly between transducers reflects at numerous points and is not dissipated before the measurement signal arrives. This interference signal completely masks the desired signal as shown in Figure 4.28.

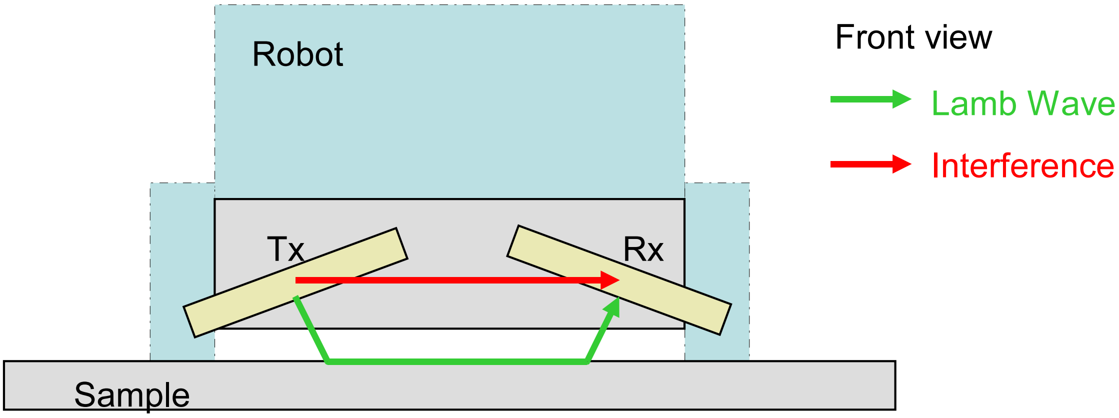


Figure 4.27: A schematic diagram of two ultrasonic transducers mounted on a Remote Sensing Agent. In addition to the intended path, the ultrasonic energy can travel from the transmitter, through the mounting to the receiver. This masks the desired signal.

The mechanical coupling should remain constant, whereas the Lamb wave signal will vary. In theory, the interference signal could be recorded before inspection and subtracted from the measurements to reveal the desired signal. However, it was found that the interference signal was too inconsistent to provide acceptable performance. Another option that was considered, was to try to dissipate the directly coupled signal using lossy material, but this approach complicated the mechanical design, made the system fragile and did not entirely remove the interference signal. The most effective solution was to design the mechanical propagation path between transducers to slow down and dissipate the ultrasonic energy as it travels directly between transducers. This was achieved by cutting slits in the transducer mounting, as shown in Figure 4.29. The slits are alternately cut from top and bottom to greatly increase the propagation distance between transducers and cause most of the energy

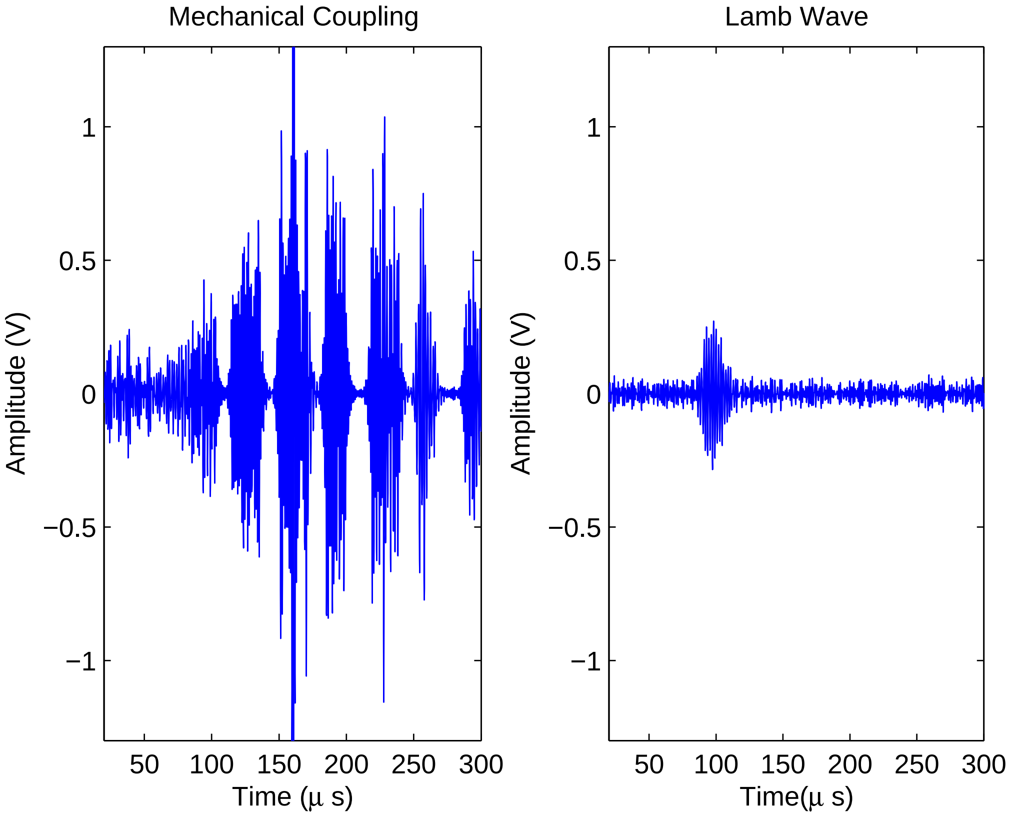


Figure 4.28: The graph on the left shows the multi-path interference issue observed with the original transducer mounting. This has been rectified in the graph on the right, which shows a clearly visible Lamb wave.

to be reflected from the slit back towards the transmitter. If the ultrasonic energy is delayed long enough, it both attenuates and arrives late enough that it can be clearly separated from the Lamb wave signal. Figure 4.28 shows the effectiveness of this approach. The graph on the left shows the received Lamb wave signal using the original mounting. The Lamb wave is masked by the interference signal. The graph on the right shows the same signal when using the mounting shown in Figure 4.29, the A_0 Lamb wave is clearly visible. The identify of the A_0 Lamb wave can be confirmed by the correct arrival time of approximately 90 μs, corresponding to approximately 2.5cm travel in air (at 340m/s) and 4cm travel in the aluminium sample(at 3000m/s for 600kHz A_0 in 1mm aluminium). Note that Figure 4.28 was captured using an early version of the receive electronics and therefore has relatively low SNR.

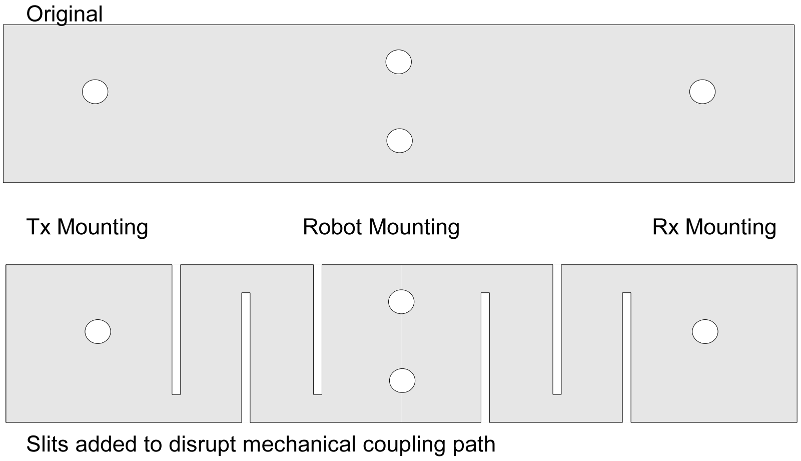


Figure 4.29: Transducer mounting to minimise direct coupling between transmitter and receiver

4.12 Characterisation and Evaluation

4.12.1 Testing dsPIC33F's Analogue to Digital Conversion

The Analogue to Digital convertor (ADC) on the dsPIC33f is used to capture the amplified receiver signal. The ADC was tested with two continuous wave signals, one at 50kHz which is considerably lower than the sample frequency (550kHz), and another at a more realistic 600kHz. A block diagram is shown in Figure 4.30, the shaded blocks are embedded on the RSA. Figure 4.31 shows the time and frequency domain representations of the captured 50kHz signal. There are 20 samples per sine wave giving a reasonable reconstruction of the signal. Figure 4.32 shows the time and frequency domain representations of the 600kHz signal. The signal was reconstructed using the algorithm discussed in Section 4.8.1.

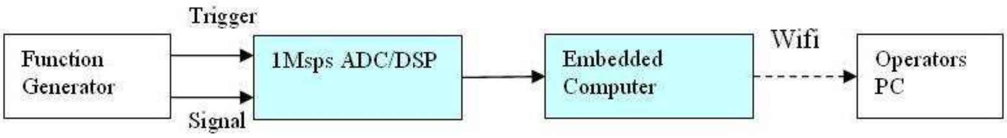


Figure 4.30: Experiment to test the system's analogue to digital convertor

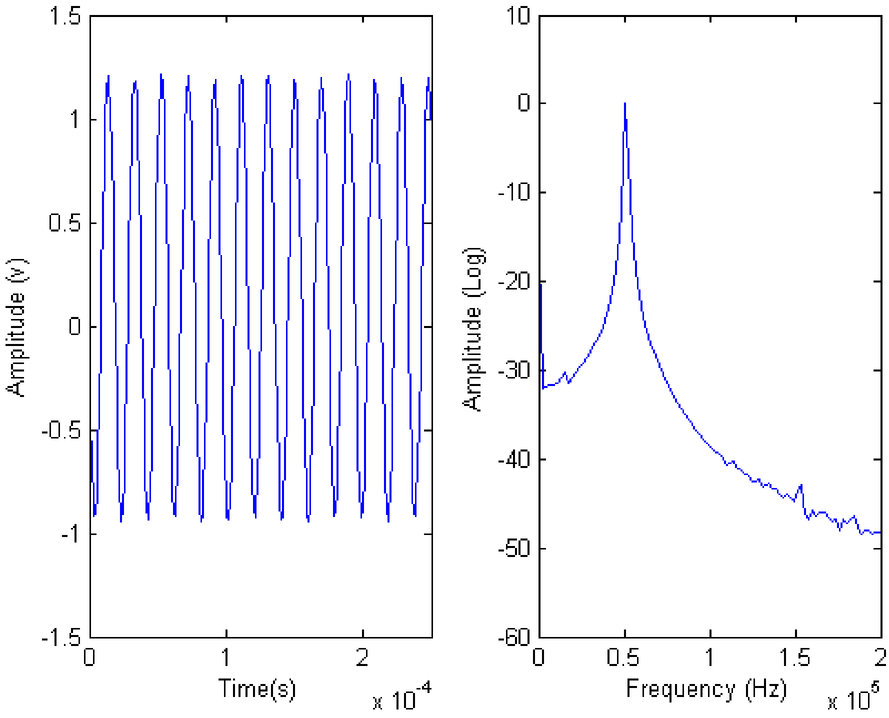


Figure 4.31: Experimentally sampled 50kHz signal

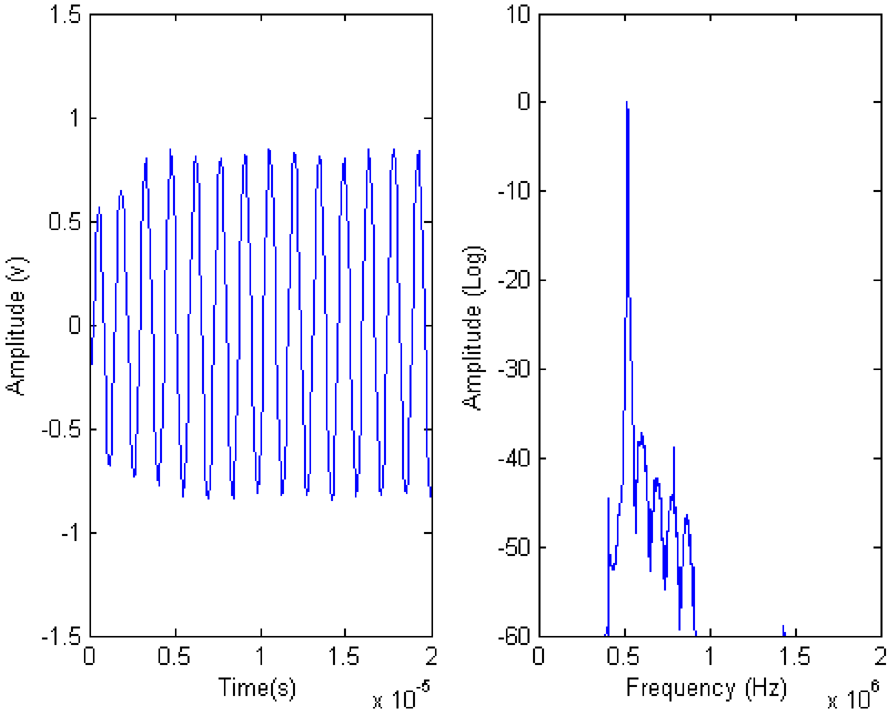


Figure 4.32: Experimentally sampled 600kHz signal

These results demonstrate good ADC performance for both low and high frequency signals. At 600kHz, the noise floor was about -40dB.

4.12.2 Testing the Complete System

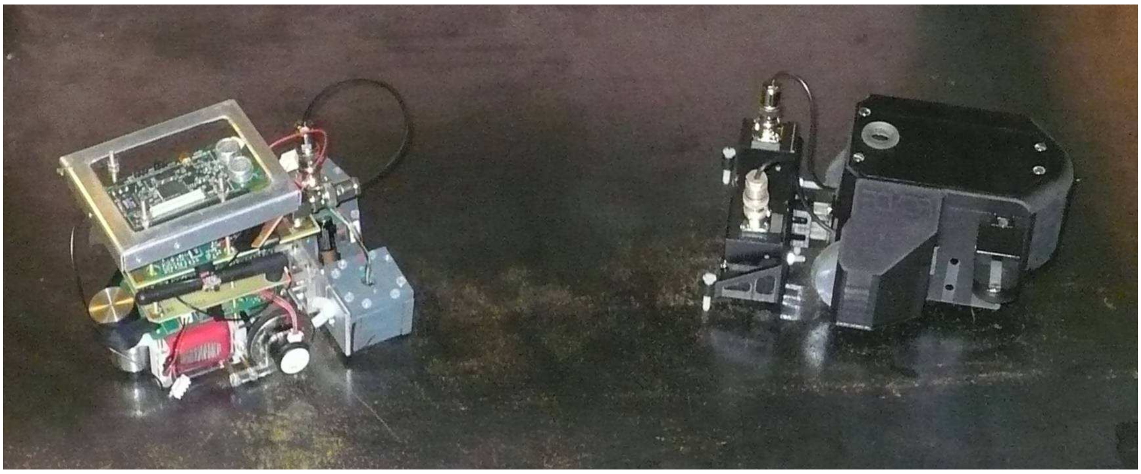


Figure 4.33: Left: Initial ultrasonic inspection robot.
Right: Robust ultrasonic inspection robot

Figure 4.33 shows the complete ultrasonic inspection robot, encapsulating all of the systems described in this chapter. A block diagram is shown in Figure 4.9.

The robot on the right has been refined to make it more robust, migrating to the new robotic platform discussed in Chapter 2. The transducer enclosures were created using the rapid prototyping machine discussed in Chapter 2. The transducers are held by a mounting that has three points of contact with the specimen. A flexible member connects the mounting to the robot, allowing it to follow the surface contour of the specimen.

Figure 4.34 shows the signal captured from a local pitch-catch experiment using a 1mm thick aluminium sample. The signal can confidently be identified as the A_0 Lamb wave mode due to its correct arrival time at $90us$, as discussed in Section 4.11.4. All systems were embedded on the robot and battery powered. The system did not need any external signal, allowing the robot to maintain its wireless capabilities. The SNR is approximately 23. Chapter 6 makes use of this payload to perform NDE.

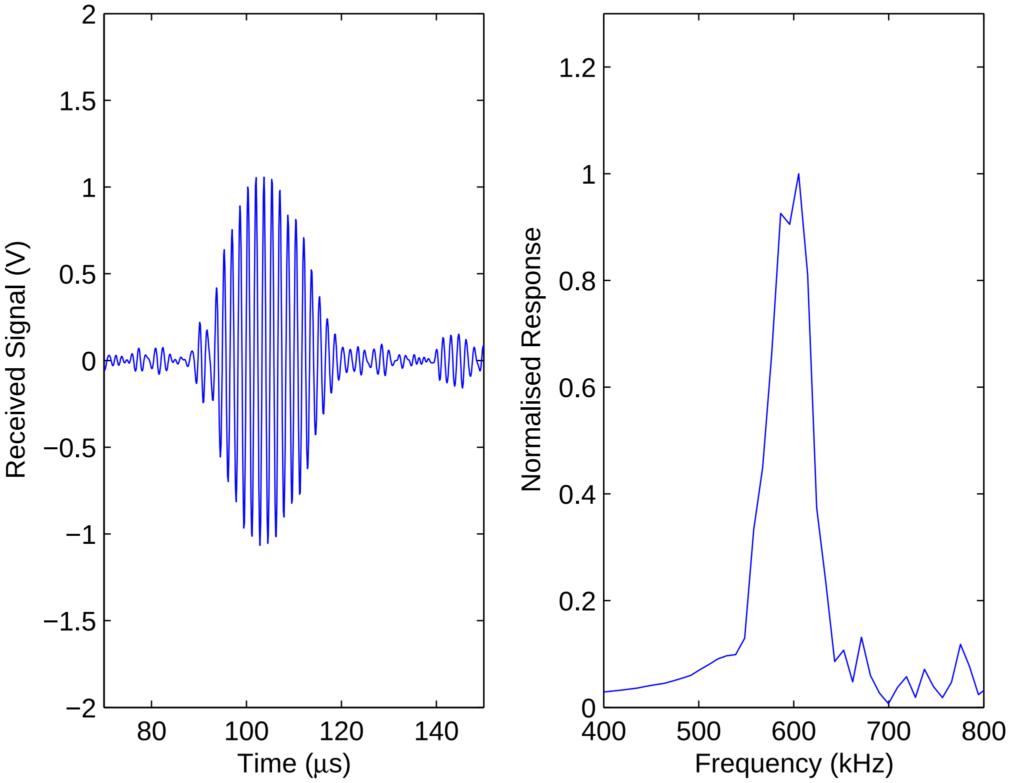


Figure 4.34: Lamb wave transmitted and measured using the embedded payload. Note that the additional waves, such as the one at $140\mu s$ are most likely to be electrical interference that has been narrow band filtered to 600kHz rather than additional ultrasonic signals.

4.12.3 Sensitivity to Surface Curvature

The coincidence principle used to generate Lamb waves using angled transducers is described in Section 4.4. The A_0 Lamb is selectively excited when there is constructive interference of the excitation signal with the Lamb wave. As the surface becomes curved the excitation signal and Lamb wave becoming misaligned, resulting in a lower amplitude Lamb wave. The experimental arrangement shown in Figure 4.35 was used to test the payload's sensitivity to curved surfaces. 1mm aluminium samples were machined to have various different curvatures. The samples were fixed onto a flat surface and the width, w and height, h of the arc were measured. The arc radius was calculated using Equation 4.12. The excitation was a 60V, 5 cycle tone burst at 600kHz. The amplitude of the pitch-catch Lamb wave was recorded for propagation over 4cm, at the optimum incident angle.

$$Radius = \frac{h}{2} + \frac{w^2}{8h} \tag{4.12}$$

The normalised receiver amplitudes are plotted against radius in Figure 4.36. As expected, the sensitivity to surface curvature is severe, with amplitude dropping off rapidly as the radius is reduced.

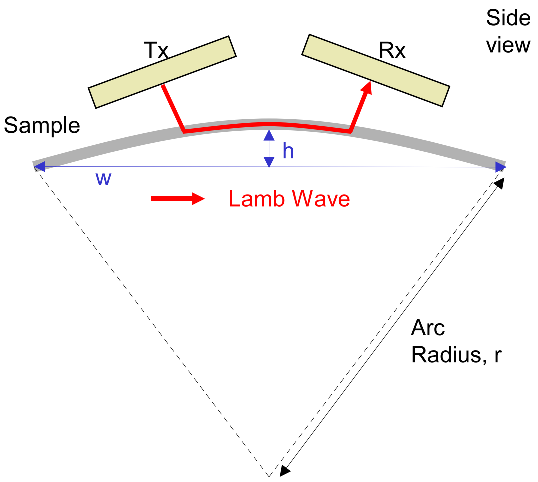


Figure 4.35: Experiment to test the ultrasonic payloads sensitivity to curved surfaces. The transducers 30 × 30 mm and were excited at 600kHz. The sample was 1mm thick aluminium.

This result has two implications for the inspection of curved surfaces such as pipes.

1. The reduction in Lamb wave amplitude will reduce the system's SNR.

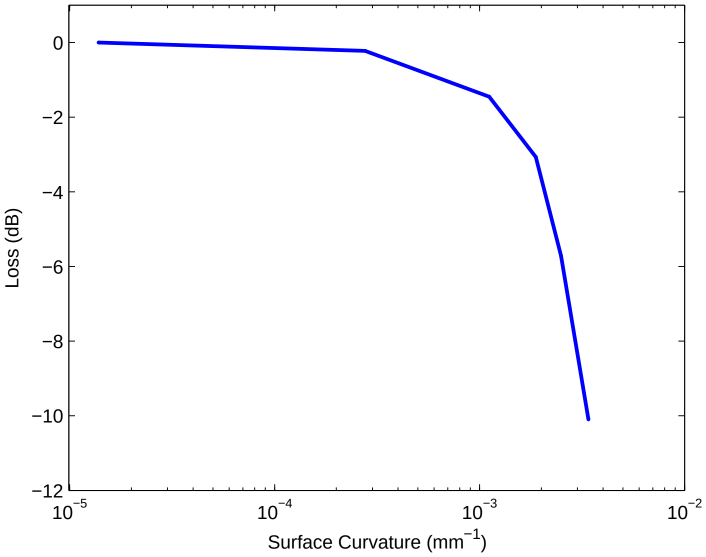


Figure 4.36: Signal attenuation against the surface curvature. Surface curvature is defined as 1 divided by the radius shown in Figure 4.35

2. If the inspection is utilising signal amplitude as the measurement, which is the case in Chapter 6, then the system will be cross sensitive to surface curvature. In addition to this, the reduction in Lamb wave amplitude is likely to be dependent on the robot's orientation relative to the surface.

Specimen curvature will have less effect on smaller transducers, since the coincidence principle only needs to hold over a smaller area. Pipes are only curved along one axis, so the transducer shape can be optimised so that the dimension tangential to the pipe circumference is kept narrow and the dimension that follows the length of the pipe is extended. However, this would require careful alignment of the transducers relative to the pipe. Alternatively, the transducer could be split into an array of separately aligned elements, but this would dramatically increase the complexity of the system.

4.12.4 Sensitivity to Material Thickness

The angled transducers work most effectively with thin plate samples. As the plate thickness is increased, more energy is required to vibrate the plate and the magnitude of the vibration and hence receiver amplitude decreases. The ultrasonic payload was tested on a series of plate thicknesses ranging from 0.75mm to 30mm. The transducers were positioned at the optimum angle for each sample, the excitation signal was a 60V, 5 cycle tone burst at 600kHz. The receiver amplitude was plotted against sample thickness in Figure 4.37. Over the range 0.75mm to 3mm, the signal amplitude decays in a fairly linear fashion. A 3mm sample yields a 11dB smaller receiver signal than a 0.75mm plate. Above 5mm the signal amplitude becomes almost constant. The frequency-thickness product is now 3MHz-mm, so the S_0 and A_0 modes have converged (See Figure 4.2) This signifies a shift from a pure A_0 Lamb wave to something closer to a Rayleigh wave. Rayleigh waves only propagate in the surface region of the material, so their amplitude is not dependent on sample thickness. It is interesting to note that the payload can propagate ultrasonic signals in a sample of any thickness, but that when the sample is thicker than 5mm, the wave only propagates in the surface region. In these thicker samples, the system will be insensitive to defects beneath the surface. This can be compensated for by lowering the operating frequency, but depending on the experiment, may result in reduced resolution (See Chapter 7 for more information).

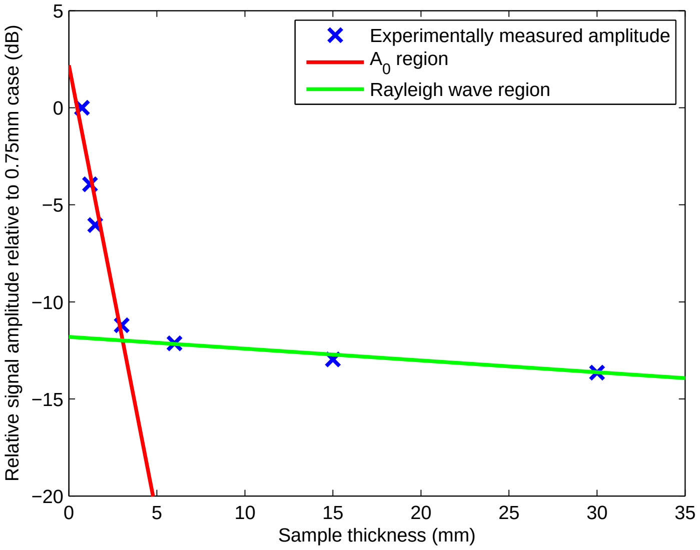


Figure 4.37: Sensitivity to plate thickness (at optimum incidence angle). The solid lines are experimental best fit lines.

4.13 Further Work

The amount of energy transmitted through the sample is proportional to the transducer surface area. Increasing the energy increases the amplitude of the measurement signals, improving SNR. However, as discussed in Chapter 6 the resolution of the measurement system is tied to the transducer dimensions. There is no optimal transducer size, merely a compromise between SNR and spatial resolution. However, there is scope to improve the systems performance, possibly by raising the excitation voltage and improving the preamplifier. This would increase SNR without increasing the transducer size. Improved SNR would facilitate some flexibility in transducer size and should allow smaller transducers with adequate SNR.

Replacing the BS108 MOSFET in the square wave amplifier with an alternative model that has lower 'on resistance' would accommodate greater excitation voltages, improving SNR. Replacing the AD8058 in the preamplifier with an ultra low noise operational amplifier would further improve the SNR. Improved SNR would make the payload better suited to smaller transducers, or thicker plate samples.

Longer term research would consider ways of improving the systems SNR by refining the piezoelectric composite matching to air or by considering other transducer technologies such as electrostatics, which are intrinsically well matched to air.

Wide bandwidth transducer would facilitate a frequency agile system that could accommodate different sample thicknesses, by changing the operating frequency. This would simplify the payload mounting mechanics. Electrostatics provide wide bandwidth, but currently do not have the sensitivity to match piezoelectric composite transducers. R. Banks has demonstrated an interesting wide bandwidth conical piezoelectric composite transducer [103], that offered both wide bandwidth and good sensitivity. This conical piezoelectric design should be investigated further.

Chapter 5

Computer Simulation of a Non-Contact Ultrasonic Payload

5.1 Introduction

The creation of a fully autonomous fleet of reconfigurable RSAs necessitates careful analysis with regard to vehicle positional accuracy, the nature of the structure under test and the optimal combination of sensor units. Arguably, this is best achieved with the aid of a computer which is capable of accurately replicating the entire system. This confers additional advantages for data interpretation and vehicle guidance. To be practically useful, full simulation in three dimensional (3D) space is required.

Commercial finite element/finite difference modelling codes such as ANSYS [116], COMSOL [117] and PzFlex [118] are capable of providing a basis for such a simulator. However, computer run-time is prohibitively expensive, especially in 3D. More approximate approaches, such as the use of ray tracing will alleviate this problem, but invariably, modelling of the underlying physics is compromised. This chapter describes an alternative approach, involving a combination of constrained sensor modelling, ray tracing and a relatively new technique, the local interaction simulation approach (LISA) [119].

This software implementation allows for the relatively straightforward integration of a simplified, but accurate 1-D model of a piezoelectric transducer [120] with the LISA wave propagation model. Additionally, it provides a potential path for the modelling to be distributed amongst the RSA platforms themselves (since each RSA contains significant on-board computational capabilities). The overall goal is to enable structurally specific inspection tasks to be optimised, taking into account both the physical aspects of the ultrasound propagation, along with the specific dynamic capabilities and restrictions of the RSA platforms. An additional complication is that in any optimisation task, it is vital to minimise the calculation speed for each individual propagation case considered. If the simulation time is too long, then effective optimisation (where many slightly different cases must be computed and compared) becomes very difficult. This area is one in which the advantages of the LISA simulation over conventional FE is highlighted.

This chapter addresses a specific form of inspection, that of air-coupled, ultrasonic Lamb wave testing of structures. From the RSA standpoint, this is an important configuration. A payload capable of preforming air-coupled, ultrasonic inspection was discussed in Chapter 4. NDE stratagies are discussed in Chapters 6 and 7. The specific contribution of this chapter was the creation and validation of a simulation tool for air-coupled angled piezoelectric composite transducers, this can be broken down as follows:

1. A C++ implementation of LISA that can be run on a desktop workstation or Strathclyde's High Performance Computer (HPC).
2. A comparison of LISA and commercial packaged PzFlex [118] for this specific application. Our implementation of LISA was found to be faster for this application (equivalent accuracy).
3. Validation of LISA for Lamb wave propagation.
4. The interface of LISA and LSM for air-coupled angled transducers simulation.

Other authors have created complete models of the air-coupled inspection problem, particularly Chimenti et al. [121, 122, 123, 124] and Castaings et al. [125, 126] who both modelled complete air-coupled inspection systems. That said, their work is generally limited to 2D geometries and does not offer the functionality (in particular rapid 3D modelling) that has been achieved herein.

5.2 Simulation Overview

Figure 5.1 shows one simulation scenario detailing the generation, propagation and reception of ultrasonic Lamb waves. An RSA simulation provides variable transmitter and receiver positions which can be used directly as the transducer locations in the ultrasonic simulation. Robot simulation was discussed in Chapter 3.

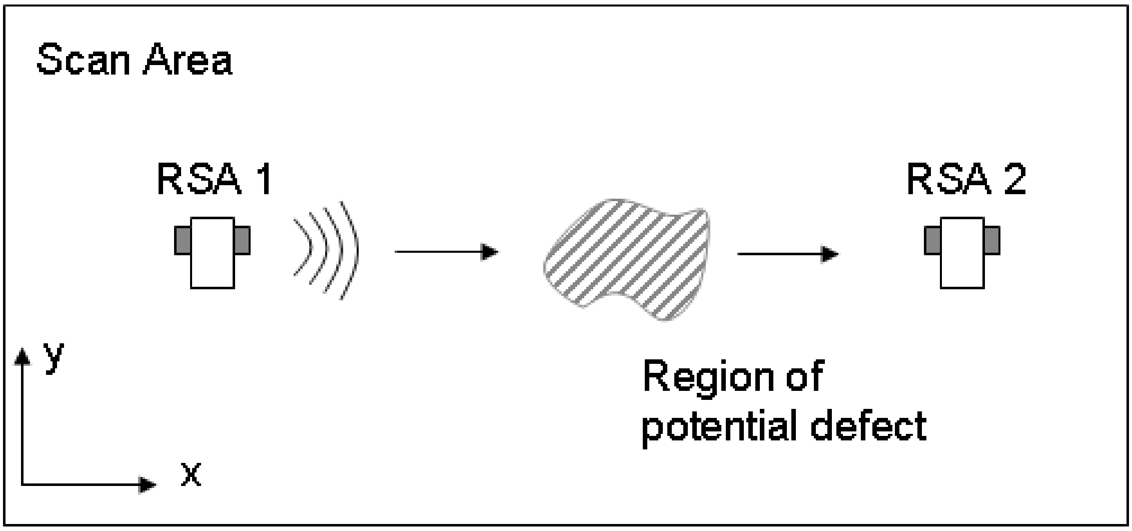


Figure 5.1: Basic Geometry for RSA scanning. Each RSA (1,2) carries an air-coupled ultrasonic transducer which can be used for transmission or reception. RSA positions are independently controlled across the inspection area.

Figure 5.2 shows a 2D representation of the ultrasonic modelling problem, illustrating the air coupled piezoelectric transducers used to generate and receive ultrasonic A_0 Lamb waves. Instead of performing a Finite Element (FE) analysis of the complete system, which would be prohibitively computationally expensive, the model is broken into sections. A Linear Systems Model (LSM) [120] is an unidimensional model of a piezoelectric ultrasonic transducer and can be used to calculate the impulse response function. A Local Interaction Simulation Approach (LISA) [119, 127, 128] is used to model wave propagation in the plate, and a ray tracing approach is adopted to interface between the two models.

Previous Lamb wave modelling work using a LISA approach [129, 130] has simplified the simulation to a 2D problem in the X-Y plane, and modelled a single Lamb wave mode as a bulk wave with the same velocity. This approach does not simulate different modes and cannot accommodate effects of mode conversion, dispersion or defects that are not full thickness. Additionally, since a X-Y model cannot model the out-of-plane displacement, it is unsuitable for the current application where the out-of-plane motion of the A_0 mode is fundamental to the measurement process.

To avoid this problem, two scenarios were considered. Firstly, a 2D X-Z plane model (considering a plate of infinite width) was used to validate the LISA model and determine the required mesh resolution. Secondly, a complete 3D model using LISA was created which allowed Lamb wave propagation in both the X and Y dimensions

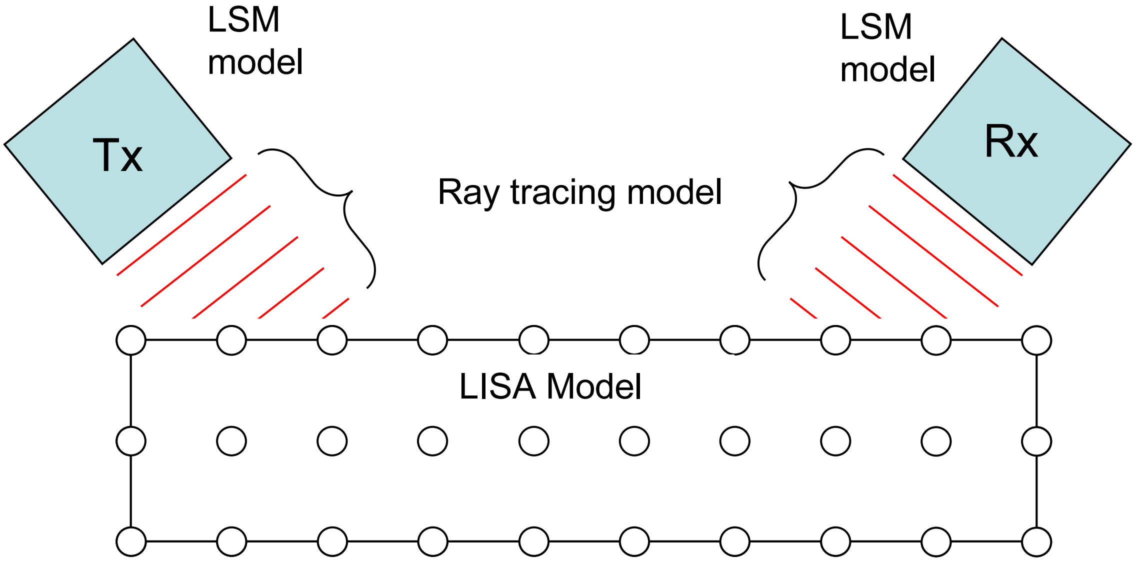


Figure 5.2: 2D representation of ultrasonic propagation simulation. Tx and Rx are modelled with linear systems approach. Plate is discretised into nodes for either FE or LISA model.

to be simulated. Since the 3D simulations were significantly more computationally expensive than their 2D counterparts, they were only performed when the test geometries considered made them essential.

5.3 Local Interaction Simulation Approach (LISA)

Finite Difference (FD) and Finite Element (FE) are standard techniques used for wave propagation modelling [131, 132, 133]. FE models are able to cope with complex geometries, but do not have the ease of implementation of FD based models. In particular, when the sample geometry is relatively simple (e.g. a rectangular plate), then FD techniques become increasingly attractive due to its ease of implementation. The LISA technique is similar to FD in that it discretises the modelling problem temporally and spatially to a series of iterative equations. With LISA, the model is created heuristically from a discretised model, whereas in FD, the partial differential equations (PDE) that describe a continuous model are discretised using FD formulation [134]. Importantly LISA bypasses the approximation made by FD when it converts derivatives into finite differences that leads to severe errors at sharp discontinuities, making LISA more accurate for inhomogeneous simulations. Delsanto et al [127] pioneered LISA development and have shown direct comparison between FD, LISA and an analytical solution for wave propagation in a bilayer.

5.3.1 Theory

This section provides an overview of the LISA theory. This section does not aim to repeat the full derivations available in Refs. [119, 127, 128], but to give the reader a flavour of the algorithm. Equations 5.11 - 5.13 were implemented in C++ to create the simulation tool.

The LISA algorithm (Equation 5.1) is relatively straight-forward in one dimension.

$$ {w_i}^{t+1} = T{w_{i-1}}^{t} + T'{w_{i+1}}^{t} - {w_i}^{t-1} \tag{5.1} $$

where

$$ T = \tfrac{2}{1+\zeta} \quad T' = \tfrac{2\zeta}{1+\zeta} \quad \zeta = \tfrac{Z_1}{Z_2} $$

This iterative equation gives the displacement of a gridpoint w_i at the next timestep $(t + 1)$, relative to the displacement of the gridpoint at the previous timestep and the displacement of the gridpoint on each side $(i + 1$ and $i - 1)$. The transmission coefficients T and T' allow for propagation through a heterogeneous medium (multilayers in the 1D case) with acoustic impedance Z. The algorithm assumes homogenous material properties in each layer, but layers can have different material properties - referred to as the Sharp Interface Model (SIM) [127]. Equation 1 assumes that the nodes are spaced for stability, a condition governed by Equation 5.2.

$$ v\tau/\epsilon <= 1 \tag{5.2} $$

where v represents the longitudinal velocity of sound in the medium, τ the timestep and ϵ the layer thickness. The reader is referred to Ref. [119] for a full derivation of the 1D algorithm.

In a 2D LISA simulation, the structure is discretised into cells, as shown in Figure 5.3. The material properties of each cell may differ, but inside the cell the material must be homogenous. Each nodal point P is at the junction of four cells. The second time derivatives across the four cells are required to converge towards a common value at the point P, which ensures that if the cell displacements are continuous at

P for the two initial times t = 0 and t = 1, they will remain continuous for all later times.

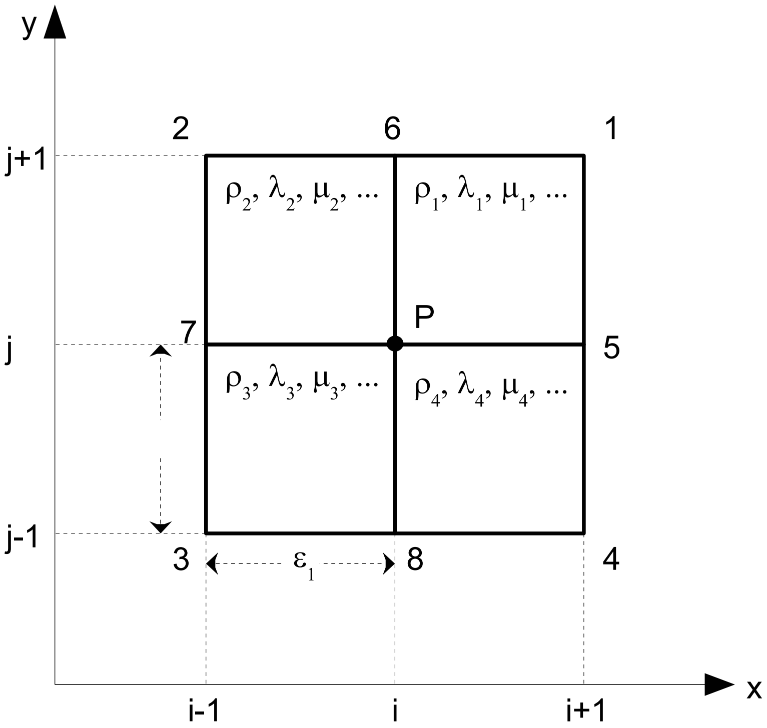


Figure 5.3: LISA spatial discretisation in 2D

Using the Sharp Interface Model [127], the material properties at P can be calculated as the average of the four surrounding cells, i.e.

$$\sigma = \frac{\sigma_1 + \sigma_2 + \sigma_3 + \sigma_4}{4} \tag{5.3}$$

$$\mu = \frac{\mu_1 + \mu_2 + \mu_3 + \mu_4}{4} \tag{5.4}$$

$$\rho = \frac{\rho_1 + \rho_2 + \rho_3 + \rho_4}{4} \tag{5.5}$$

The following definitions are made,

$$g_i = \lambda_i - \mu_i \tag{5.6}$$

$$\sigma^5 = \frac{\sigma_1 + \sigma_4}{2}, \sigma^6 = \frac{\sigma_1 + \sigma_2}{2}, \sigma^7 = \frac{\sigma_2 + \sigma_3}{2}, \sigma^8 = \frac{\sigma_3 + \sigma_4}{2} \tag{5.7}$$

λ and μ are the Lamé constants for the material. σ, λ and μ can be extracted from the material's stiffness tensor (see Ref. [127] for details). ρ is the material's density.

For stability, the Courant number, C, must be less than or equal to 1. The Courant number can be calculated from the time step, τ, cell size, ϵ and the material's longitudinal and transverse velocities v_L and v_T. This is the 2D generalisation of Equation 5.2.

$$C = \frac{\tau\sqrt{v_L^2 + v_T^2}}{\epsilon} \leq 1 \tag{5.8}$$

The particle displacements in the x and y directions in Figure 5.3 are denoted as u and v respectively. Note that $v_i = \lambda_i + \mu_i$, these u and v displacements at the point P can be calculated as,

$$\begin{aligned} u_{t+1} = \\ & 2u_t - u_{t-1} + \frac{1}{\rho}[\sigma_5{}^{(1)}u_5 + \sigma_7{}^{(1)}u_7 + \mu_6 u_6 + \mu_8 u_8 \\ & - 2(\sigma^{(1)} + \mu)u_t - \frac{1}{4}\sum_{k=1}^{4}(-1)^k v_k v_t \\ & - \frac{1}{4}\sum_{k=1}^{4}(-1)^k v_k v_k + \acute{g}_5 v_5 + \acute{g}_6 v_6 + \acute{g} v_7 + \acute{g}_8 v_8] \end{aligned} \tag{5.9}$$

$$\begin{aligned} v_{t+1} = \\ & 2v_t - v_{t-1} + \frac{1}{\rho}[\mu_5 v_5 + \mu_7 v_7 + \sigma_6{}^{(2)}v_6 + \sigma_8{}^{(2)}v_8 \\ & - 2(\sigma^{(2)} + \mu)v_t - \frac{1}{4}\sum_{k=1}^{4}(-1)^k v_k u_t \\ & - \frac{1}{4}\sum_{k=1}^{4}(-1)^k v_k u_k + \acute{g}_5 u_5 + \acute{g}_6 u_6 + \acute{g} u_7 + \acute{g}_8 u_8] \end{aligned} \tag{5.10}$$

where: $\acute{g}_5 = \frac{1}{2}(g_4 - g_1)$, $\acute{g}_6 = \frac{1}{2}(g_1 - g_2)$, $\acute{g}_7 = \frac{1}{2}(g_2 - g_3)$ and $\acute{g}_8 = \frac{1}{2}(g_3 - g_4)$

Equations 5.9 and 5.10 are the principal displacement equations of LISA in two dimensions. A derivation can be found in [127], although note that a number of small errors present in the formulae within that paper have been corrected in [129] and [130]. These two equations rely solely on known material properties σ, λ and μ for each cell and arbitrary discretisation steps, both spatial and temporal.

The LISA iteration equation can be extended to 3D geometries [128]. The 3D representation is described in Figure 5.4. Each node is surrounded by eight cells and eighteen nodes. The cells may have different mechanical properties, but the properties inside each cell are homogenous. The cells are referenced by their position relative to centre node, O, using the parameters α, β and γ which represent sub-cell deviations in the x, y, and z axis respectively. For example, $\sigma_1(\alpha = -, \beta = -, \gamma = -)$ refers to σ_1 in the front, left cell, highlighted in red in Figure 5.4. The cell size and time step required for stability can be calculated using Equation 5.8.

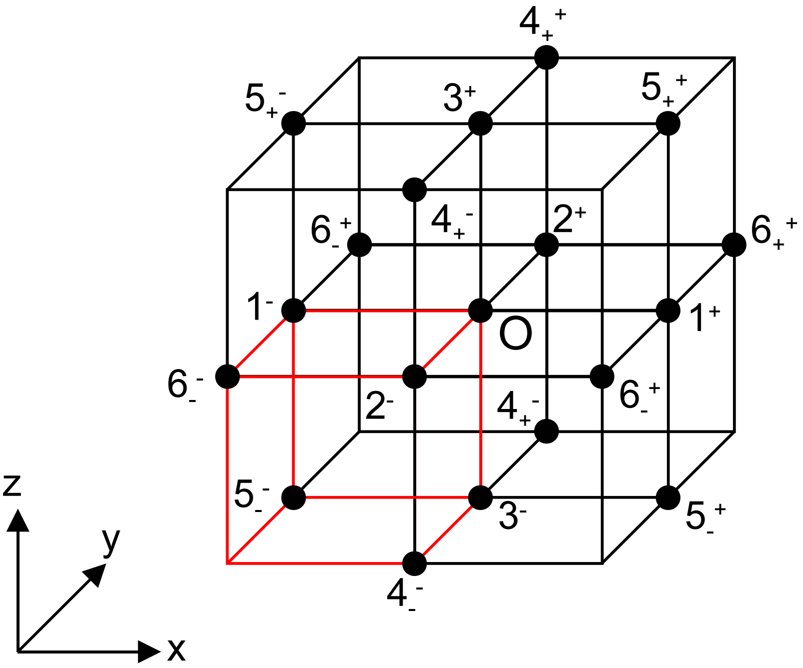


Figure 5.4: LISA spatial discretisation in 3D

The displacement at the node O, at the current time step, can be calculated from the displacement of node O at current and previous time steps, and the displacement of the eighteen surrounding nodes at the current time step. A derivation can be found in Ref. [128], the equations are repeated here for completeness. Again the material properties σ, λ and μ are extracted from the material's stiffness tensor (see Ref. [128] for details).

$$
\begin{aligned}
u_{t+1} = &\\
&-u_{t-1} + 2u\left[1 - \chi\left(\bar{\sigma}_1 + \bar{\mu}_5 + \bar{\mu}_6\right)\right]\\
&+\frac{1}{8}\chi \sum_{\alpha,\beta,\gamma=\pm} \{2\tilde{\sigma}_1 u(1^\alpha) + 2\tilde{\mu}_6 u(2^\beta) + 2\tilde{\mu}_5 u(3^\gamma)\\
&\quad +\alpha\beta\left[(\tilde{v}_6(v(6^\alpha_\beta) - v)) + \tilde{\phi}_6(v(2^\beta) - v(1^\alpha))\right]\\
&\quad +\alpha\gamma\left[(\tilde{v}_5(z(5^\alpha_\gamma) - z)) + \tilde{\phi}_5(z(3^\gamma) - z(1^\alpha))\right]\}
\end{aligned}
\tag{5.11}
$$

$$
\begin{aligned}
v_{t+1} = &\\
&-v_{t-1} + 2v\left[1 - \chi\left(\bar{\sigma}_2 + \bar{\mu}_4 + \bar{\mu}_6\right)\right]\\
&+\frac{1}{8}\chi \sum_{\alpha,\beta,\gamma=\pm} \{2\tilde{\sigma}_2 u(1^\alpha) + 2\tilde{\mu}_4 v(2^\gamma) + 2\tilde{\mu}_6 v(1^\alpha)\\
&\quad +\alpha\beta\left[(\tilde{v}_6(u(6^\alpha_\beta) - u)) + \tilde{\phi}_6(u(1^\alpha) - u(2^\beta))\right]\\
&\quad +\beta\gamma\left[(\tilde{v}_4(z(4^\beta_\gamma) - z)) + \tilde{\phi}_4(z(3^\gamma) - z(2^\beta))\right]\}
\end{aligned}
\tag{5.12}
$$

$$
\begin{aligned}
z_{t+1} = &\\
&-z_{t-1} + 2z\left[1 - \chi\left(\bar{\sigma}_3 + \bar{\mu}_4 + \bar{\mu}_5\right)\right]\\
&+\frac{1}{8}\chi \sum_{\alpha,\beta,\gamma=\pm} \{2\tilde{\sigma}_3 u(3^\gamma) + 2\tilde{\mu}_4 z(2^\beta) + 2\tilde{\mu}_5 z(1^\alpha)\\
&\quad +\beta\gamma\left[(\tilde{v}_4(v(4^\beta_\gamma) - v)) + \tilde{\phi}_4(v(2^\beta) - v(3^\gamma))\right]\\
&\quad +\alpha\gamma\left[(\tilde{v}_5(u(5^\alpha_\gamma) - u)) + \tilde{\phi}_5(u(1^\alpha) - u(3^\gamma))\right]\}
\end{aligned}
\tag{5.13}
$$

where

$$
\chi = \frac{\delta^2}{\rho\epsilon^2}, \quad \phi_n = \lambda_n - \mu_n, \quad \tilde{\sigma}_k = \sigma_k\left(\alpha,\beta,\gamma\right), \quad \bar{\sigma}_k = \frac{1}{8}\sum_{\alpha,\beta,\gamma} \tilde{\sigma}_k,
$$

A simulation of wave propagation in an aluminium plate may consist of a layer of cells surrounded by a thin layer of air. The material properties can be defined for each cell, so defects, such as slots, can be represented by replacing some of the plate cells with air cells. The fundamental simplicity of the LISA algorithm allows for highly efficient implementation. The LISA discretisation is regular, so the locations of cells do not need to be stored in memory, but can be calculated at run time, further improving speed whilst reducing memory requirements. Since each node has to be individually calculated, LISA does not scale particularly well - as volume (V) increases, the number of nodes increases by V^3. However this is a general problem with discritisation in FD and FE models and not specific to LISA. Importantly, since each node only depends on its nearest neighbours, the model lends itself well to parallelisation. This opens the way for future work to potentially implement the LISA modelling task itself across a fleet of RSA vehicles each with its own microcontroller based acquisition and control system.

5.4 LISA Validation

In order to validate the LISA propagation model, a comparison between LISA, standard FE modelling software and numerical solution to the Rayleigh-Lamb frequency equations was performed. Guided waves were chosen for the validation exercise as the presence of multiple modes of propagation, combined with the variation of phase velocity with Frequency Thickness Product (FTP), provided an exacting test of the accuracy of the results.

5.4.1 Dispersion

An approach [135] based on a spatially sampled impulse response function, followed by a 2D FFT was used to recover the spatial and temporal frequency components of the propagating waves. A 2D simulation of a 250mm long, infinitely wide, 1mm thick aluminium plate was performed. The plate was excited with a single cycle of a sine wave ($2\mu s$ duration with 0.125mm spatial diameter) and spatially sampled to create a series of discrete time surface displacement measurements as shown in Figure 5.5. The excitation was applied to a distance of 1/3 into the plate (83mm) and the simulation was run only until the wave reached the outer edge of the plate to ensure no edge reflections were present. Taking a 2D FFT of this time-space data matrix produced frequency-wavenumber [f-k] space data which was plotted to reveal the dispersion of the propagating waves; as shown in Figure 5.6.

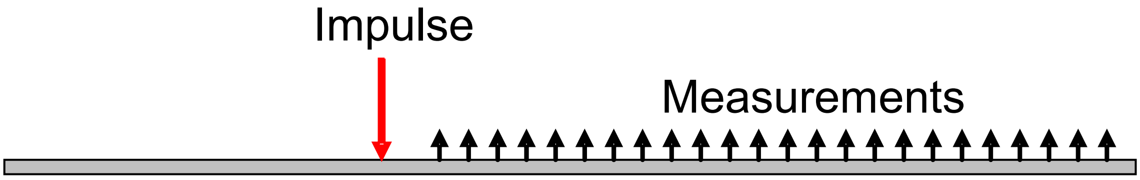


Figure 5.5: Simulation of spatially sampled broadband wave propagation in a plate

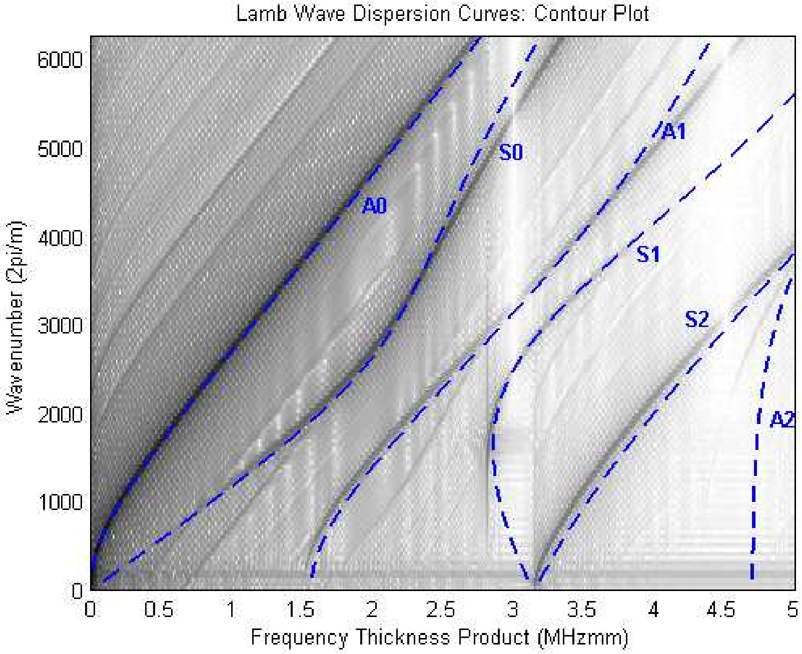


Figure 5.6: Contour plot of Lamb wave dispersion matrix: LISA simulation results for 2D model, plate thickness 1mm, 8 cells

Figure 5.6 shows a contour plot of the [f-k] dispersion matrix, which exhibits good agreement with analytical theory [90] shown as dashed lines. The multiple modes are clearly visible and of the correct shape. This plot was generated using a 2D LISA model where the 1mm thickness was divided into 8 cells. To compare the accuracy of fit of the simulation with the numerical solutions of 'Disperse' [90], a method of extracting the local maxima of the dispersion curves was developed, followed by calculation of an error function. A filter was written to automatically extract the local maxima in the [f-k]dispersion plot. The filter considered each pixel in the image and subtracted the mean of the surrounding N^2 pixels (N = 14 was found to work best). It then normalised the data and set any point with a value below 50% to 0 and anything above 50% to 1. Finally, the filter replaced any small clusters of points with a single point equal to the average.

Figure 5.7 shows the filter applied to the data of Figure 5.6 to extract the local maxima.

In order to have an objective measure of the goodness of fit, the normalised mean-square error (MSE) is introduced, the definition is:

$$MSE(\hat{x}) = \frac{100}{N\sigma_x^2}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2 \tag{5.14}$$

where the caret denotes an estimated quantity. This MSE has the following useful property; if the mean of the output signal $\overline{x}$ is used as the model i.e. $\hat{x}_i = \overline{x}$ for all i, the MSE is 100.0, i.e.

$$MSE(\hat{x}) = \frac{100}{N\sigma_x^2}\sum_{i=1}^{N}(x_i - \overline{x})^2 = \frac{100}{\sigma_x^2}.\sigma_x^2 = 100$$

Experience shows that a MSE of less than 5.0 indicates good agreement while one of less than 1.0 reflects an excellent fit.

The accuracy of LISA was measured by calculating the normalised mean squared error of dispersion curves using 'ideal' data generated numerically by Disperse [90]. $\hat{x}_i$ is the frequency thickness product of a sample from the LISA dispersion plot and x_i is the corresponding point from Disperse for the same wavenumber (using linear interpolation between discrete data points where required).

5.4.2 Cell Density Considerations

One of the most significant parameters in the implementation of LISA is the mesh density. Increasing the density improves the accuracy, but increases the computational complexity of the model. In 2D simulations, fine meshes do not present such a problem, as most simulations take less than a few minutes to run. However in 3D, the computation is more involved and run times can reach several hours. It should be noted that to maintain stability, halving the cells size also requires the time step to be halved, as indicated in Equation 5.2. In 3D, halving the time step increases the run time by the power law t^4 (double the number of cells in each dimension plus double the total number of frames). Memory requirements scale linearly with the number of cells, making large 3D simulations prohibitive on desktop workstations. Mesh Sensitivity Analysis showed that reducing the cell size (hence increasing the density) improved the correlation with theory. The results are shown in Figure 5.9. Note that the cell density was specified in cells/mm rather than cells/wavelength as no specific operating frequency/wavelength was being considered at this point.

It should be noted that since the current RSA platforms selectively generate and receive just the A_0 mode, good correlation with A_0 is likely to be sufficient for current applications. The minimal functional density is 2 $cells/mm$, which gives a reasonable approximation of A_0 with a MSE of 1.6, however no other modes are present. Increasing to 4 cells/mm provides an excellent approximation of A_0 (MSE = 0.13) and a reasonable approximation of S_0 (MSE =4.3), this is shown in Figure 5.8, noting that S_0 diverges from the analytical solution for FTP $>$ 2 MHzmm. In order to get an excellent match for both A_0 and S_0 (MSE $<$ 1) the cell density must be increased to 8 cells/mm.

5.5 Comparison with a Commercially Available Package

The low level control provided by custom implementation facilitates interfacing with other models, such as the LSM, that would not be straightforward with a commercial closed-source package. However, in order to fully justify the use of LISA over commercial finite element packages, a comparison was performed between LISA and a leading simulation package PZFlex [118] to compare accuracy and calculation speed for this specific application.

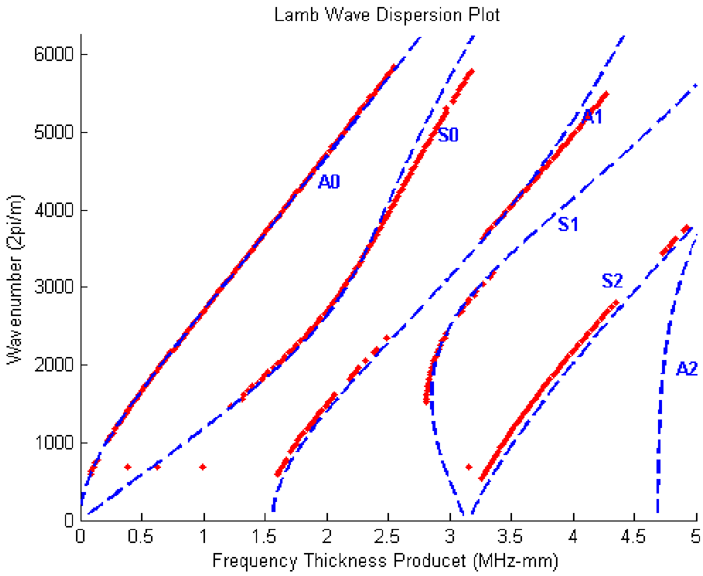


Figure 5.7: Comparison between automatic local maxima extraction algorithm and numerical solution (dashed lines). Simulation had 8 cells per mm.

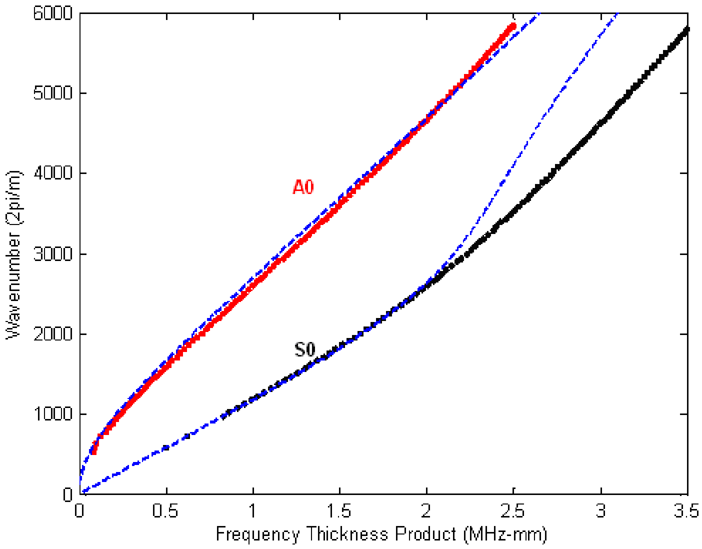


Figure 5.8: LISA simulation of 1mm thick plate with 4 cells through the thickness - showing the S_0, A_0 modes and analytical solution (dashed lines)

5.5.1 Accuracy

As discussed in Section 5.2, a 3D simulation is required for high accuracy. However as discussed in Section 5.4.2 the computational intensity of 3D models often results in prohibitively long run times and hardware requirements. This makes a coarser mesh with a reduction in accuracy a practical compromise. As expected PzFlex followed the same trend. The publishers of PzFlex recommend a minimum of 20 cells/mm for Lamb wave plate simulation, however this is impractical in 3D, so a mesh sensitivity study was performed in 2D to determine how the accuracy degraded with the cell density. The results for PzFlex are shown in Figure 5.10. It was found that below 6 cells/mm PzFlex did not produce coherent results and that 6 cell/mm provided a good match for A0 and a reasonable match for S0 (MSE = 0.16 and 3.3 respectively) - Figure 5.11 which should be compared with the 4 cells/mm result for LISA shown in Figure 5.8.

5.5.2 Speed

Using LISA with 4 cells/mm through the thickness provided a good tradeoff between speed and accuracy and was roughly equivalent to 6 cells/mm for PzFlex. Both provide a good approximation of A_0 and a reasonable approximation of S_0 which diverges at higher wavenumbers, as shown in Figures 5.8 and 5.11.

In order to compare the simulation speed, a 250 × 250 × 1mm aluminium plate was modelled in 3D using 4 cells/mm for LISA and 6 cells/mm for PzFlex. The simulation was identical to that preformed in Section 5.4.1 except the 3D geometry was modelled. The simulations were set to run for $90us$ of simulation time. The results are shown in Table 5.1. The comparison was performed on a Windows XP (64 bit edition) 2.4GHz quad-core workstation (2 dual AMD Opterons) with 16GB RAM.

Table 5.1: PZFlex vs LISA speed comparison

Package	Time to run 90us simulation
PzFlex	2 hours 58 minutes
LISA	53 minutes

It is clear that in this instance the custom LISA propagation code outperformed the commercial package by a factor of approximately 3.

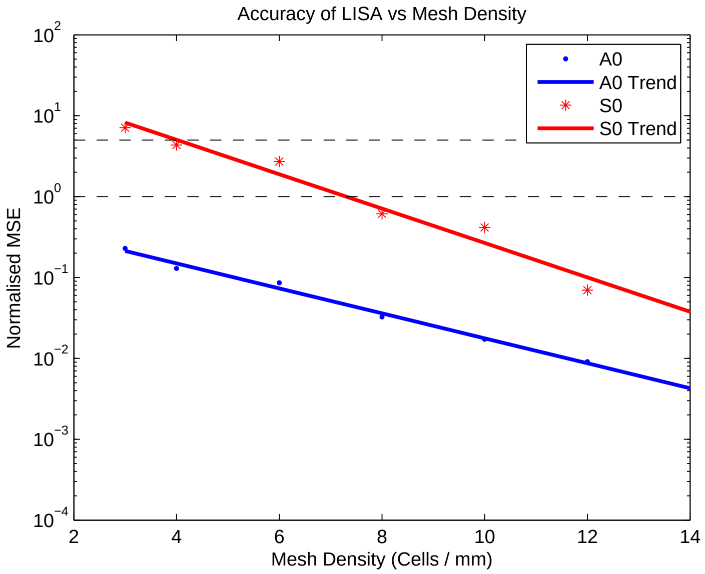


Figure 5.9: Variation of LISA accuracy (2D) as a function of mesh density expressed as MSE from analytic model. The horizontal dashed lines represent MSE = 5 - good agreement and MSE = 1 - excellent agreement

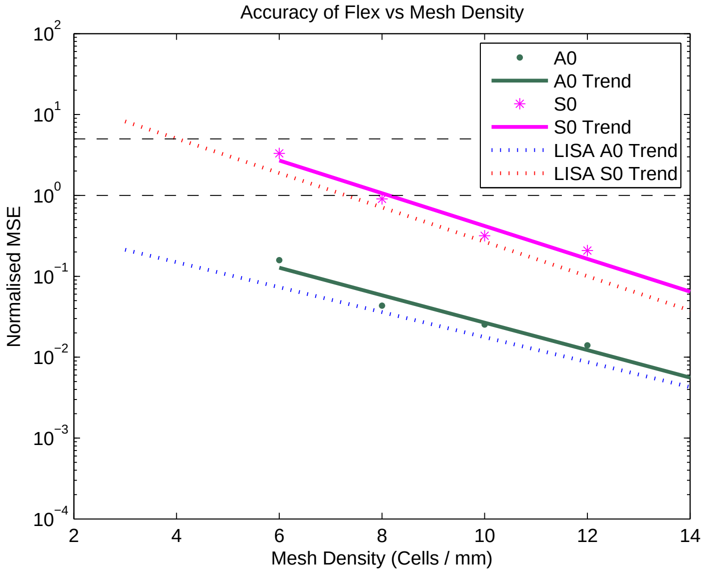
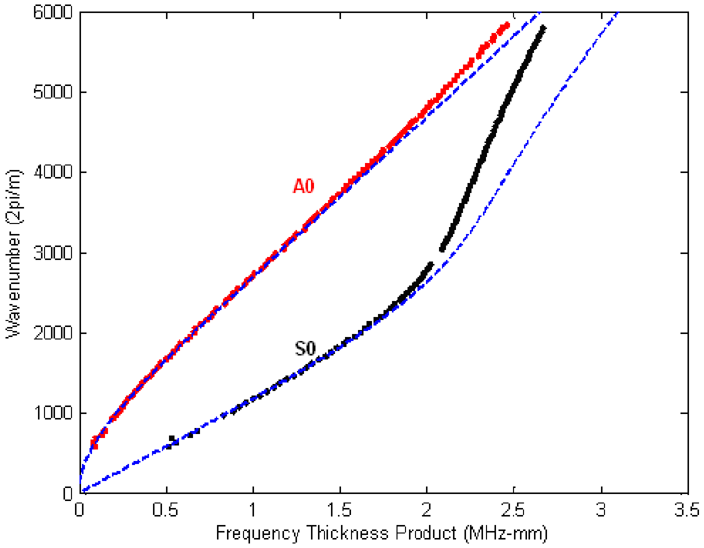


Figure 5.10: Accuracy comparison (2D) between LISA and PzFlex as function of mesh density. The horizontal dashed lines represent MSE = 5 - good agreement and MSE = 1 - excellent agreement



hp]

Figure 5.11: PzFlex simulation of 1mm thick plate with 6 cells though the thickness - showing A_0, S_0 modes and analytic solution (dashed line)

5.6 Angled Transducers

This section links the LSM with the LISA wave propagation model to create an angled transducer pitch-catch model, to simulate air-coupled Lamb wave generation and detection, as required by the RSAs.

5.6.1 Linear Systems Model

A Linear Systems Model [120] was used to model the response of the transmitter and receiver transducers. The input drive or received signal was convolved with the simulated transducer impulse response. This approach is discussed in detail in Refs [136], [137]. The transducer designs were based on the payload discussed in Chapter 4. The simulation considered 1-3 piezocomposite transducers that were designed to operate at a centre frequency of 600kHz in pitch-catch mode. The transmitter had a 70% volume fraction of PZT-5H and the receiver had a 30% volume fraction of PZT-5A. In both cases the passive filler material was epoxy (CY1301/HY1300). A silicon based, low-loss matching layer was integrated onto the front-face of each transducer to minimise insertion loss due to the impedance mismatch between the transducer face and air. This is discussed further in Chapter 4.

5.6.2 A_0 Lamb Wave Generation

When the transmitter is excited, the axial mode produces a planar wave radiating from the front face (as shown in Figure 5.12). Efficient matching ensures most of the energy radiates from the front of the transducer [104]. Also, if a 1-3 piezocomposite is used, the composite nature of the transducer helps to dampen out unwanted radial modes. The angle θ was selected to phase match the transducer output to the desired Lamb wave mode as described by Snell's Law. For propagation in a 1mm thick aluminium plate at 600 kHz, the appropriate value of $\theta_i = 9.8^o$ ($c_{ph} \simeq 2000$ m/s).

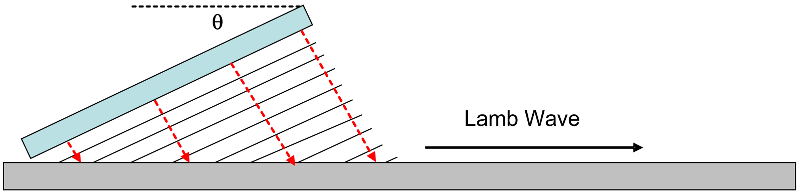


Figure 5.12: Air-coupled generation of A_0 Lamb mode in aluminium sample plate

A ray tracing technique was used to model the excitation of the specimen under test. For each LISA node under the transmitter, an excitation was applied at an angle perpendicular to the transmitter. The air channel was modelled as a delay (assuming no attenuation). Nodes near the base of the transducer were excited first, with the delay increasing with distance from the base. Note it could take several cycles before the whole area under the transducer was excited. Incorporating a model of the near field [138] fell outwith the scope of this work, but the loss in accuracy is minimal since slight variations in the amplitude across the excitation wave are averaged over the Lamb wave as it passes through the excitation region. Since the transducer beam interferes with the plate well inside the near field, beam spread is negligible.

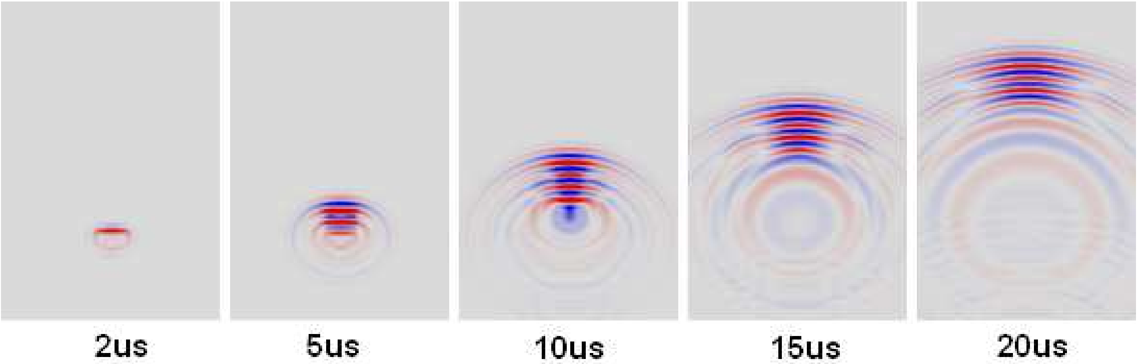


Figure 5.13: Simulation of air-coupled A_0 generation in a 1mm thick aluminium plate

Figure 5.13 shows an example simulation of a 10mm circular transducer placed over a 1mm aluminium plate, the Figure illustrates the amplitude of the out-of-plane displacement. The transducer was excited with a 600kHz, 3 cycle tone burst and was set at an angle of 9.8^o facing forward (angle in XY plane = 0^o). The A_0 wavepacket can be clearly seen.

5.6.3 A_0 Lamb Wave Measurement

Snell's law works reciprocally, so for matched reception, the angle of the receiver must be the same as that of the transmitter. This allows the wave to recombine constructively on the receiver's face. Again a ray tracing approach was adopted, where only nodes that fell under the transducer's field of view were considered. For each node the component of displacement normal to the receiver face was added to the receiver with a time delay corresponding to the perpendicular distance between the node and transducer face. At each time-step the transducer model integrated over all inputs to produce a single output.

Figure 5.14 shows a plot of frequency sensitivity of a wide-band receiver as a function of receiver angle. The simulation consisted of a 1mm thick plate excited by an impulse. A receiver was positioned 100 mm away from the source. The angle of the receiver was incrementally increased from 0^o to 30^o in steps of 0.2^o and for each angle the response in the frequency domain was recorded. The frequency domain intensity for each angle is shown in grayscale. A line plot was overlaid to show the theoretical result calculated from Snell's law [9], taking phase velocities from the theoretical dispersion curve of the A_0 Lamb wave mode. It is clear that the receiver model response was well matched to the theoretical curve for the A_0 mode.

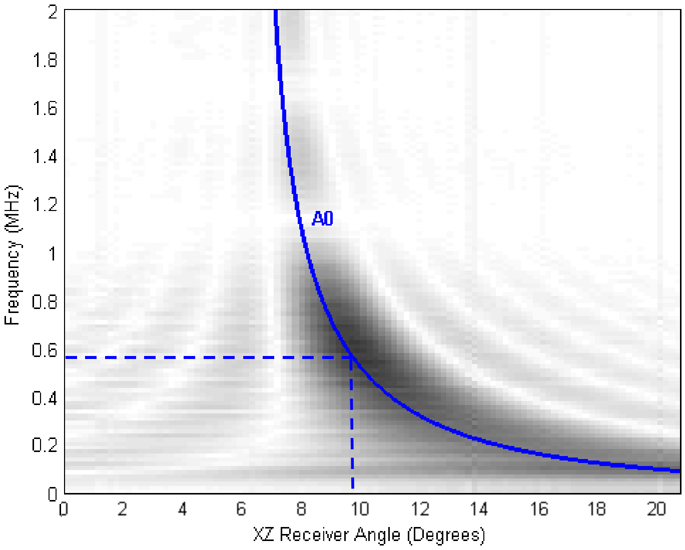


Figure 5.14: Combined ray tracing and LISA simulation of angular response of air-coupled transmitter. Snells's law solution shown as solid line

5.7 Interfacing LSM and LISA

The LISA model simulates the displacement field generated by elastic wave propagation in the sample under investigation, whereas the LSM approximates the pressure field output from a transmitter excited by a voltage, or the voltage generated by the incident wave on the receiver's face. In order to interface the two models, the excitation pressure wave must be converted to a displacement excitation for LISA and the resulting displacement wave under the receiver must be converted back to a pressure wave. Equation 5.15 can be used to perform this conversion, where Z is the acoustic impedance of the plate and v_{node} is the velocity of the point in question.

$$P = Zv_{node} \tag{5.15}$$

v_{node} can be calculated by differentiating the displacement w.

$$P_t = Z(\frac{w_t - w_{t-1}}{\tau}) \tag{5.16}$$

The last factor that has to be considered is the impedance mismatch between the plate and surrounding air. The transmission coefficient of a plane wave at normal incidence can be estimated from Equation 5.17.

$$T_P = \frac{P_t}{P_i} = \frac{2Z_2}{Z_1 + Z_2} \tag{5.17}$$

P_t and P_i are the transmitted and incident pressure waves respectively travelling into a material with acoustic impedance Z_2 from a material with acoustic impedance Z_1 at normal incidence. The excitation pressure used to calculate the excitation displacement in Equation 5.15 is calculated by multiplying the output from the transmitter LSM by the transmission coefficient for air to aluminium. The pressure used as the input to the receiver LSM is first calculated by Equation 5.15 then multiplied by the transmission coefficient for aluminium to air.

5.7.1 Simulation Results

Figure 5.15 shows a full simulation of a time domain pitch-catch experiment between angled transducers, incorporating both LISA and the Linear Systems Transducer model compared with experimental measurements. The excitation signal was a 3 cycle tone burst of 10V at 600kHz. The separation between the transducers was 100mm. The transducers were identical to those described in Section 5.6.1 and had an active area of 30mm × 30mm. The transmitter was driven directly from the 50Ω output of an Agilent 33220A function generator. The receiver was amplified by 40dB using an Olympus 5670 preamplifier and connected to the 1MΩ input of an Agilent 54624A oscilloscope. The simulated result has a slightly narrower bandwidth, this is most likely due to a simplification in transducer modeling that ignores loss.

Figure 5.16 shows five, 3D pitch-catch simulations, between a 10mm diameter angled transmitter and receiver. For clarity, the diagrams are drawn in 2D. The plate was 125mm × 125mm × 1mm. The transmitter was positioned at x = 62.5mm, y = 25mm at an angle of 9.8 degrees to the surface. In case I a point probe was placed to measure the out-of-plane displacement on the top surface at x = 62.5mm, y = 100mm. In case II an angled receiver was placed in the same location as the probe positioned to face the transmitter and held at 9.8 degrees (XZ). In case III, the receiver was rotated to face away from the transmitter. Case IV and V are the same as case II, with the addition of void regions in the model to simulate defects. In Case IV the defect was a 0.5 mm deep surface breaking void which was 3mm long in the Y direction, it was 10mm wide in the X direction and positioned 80mm in from the plate edge 'C' (62.5mm, 80mm and 0.75mm in X, Y and Z respectively). In Case V the defect was extended to be 10mm long in the Y direction and was positioned in the centre of the plate thickness (62.5mm, 80mm and 0.5mm in X, Y and Z respectively).

Figure 5.17 shows the normalised output for each case (in cases III - IV, the amplitudes are normalised against case II). In case I, four wave packets are visible which represent the initial wave passing under the probe (A-B), the reflection from the back edge (A-C-B), the reflection from the back edge reflected off the front edge beside the transducer (A-C-D-B) and finally the packet reflected again off the back edge (A-C-D-C-B). The probe was assumed to be ideal, so only the transmitters transfer function was included. In cases II - V a narrow band piezocomposite receiver was used which causes the signals to take longer to decay. In case II only the first and third packets are visible since the receiver was only sensitive to incident waves. This directional sensitivity is also apparent in case III where only the back

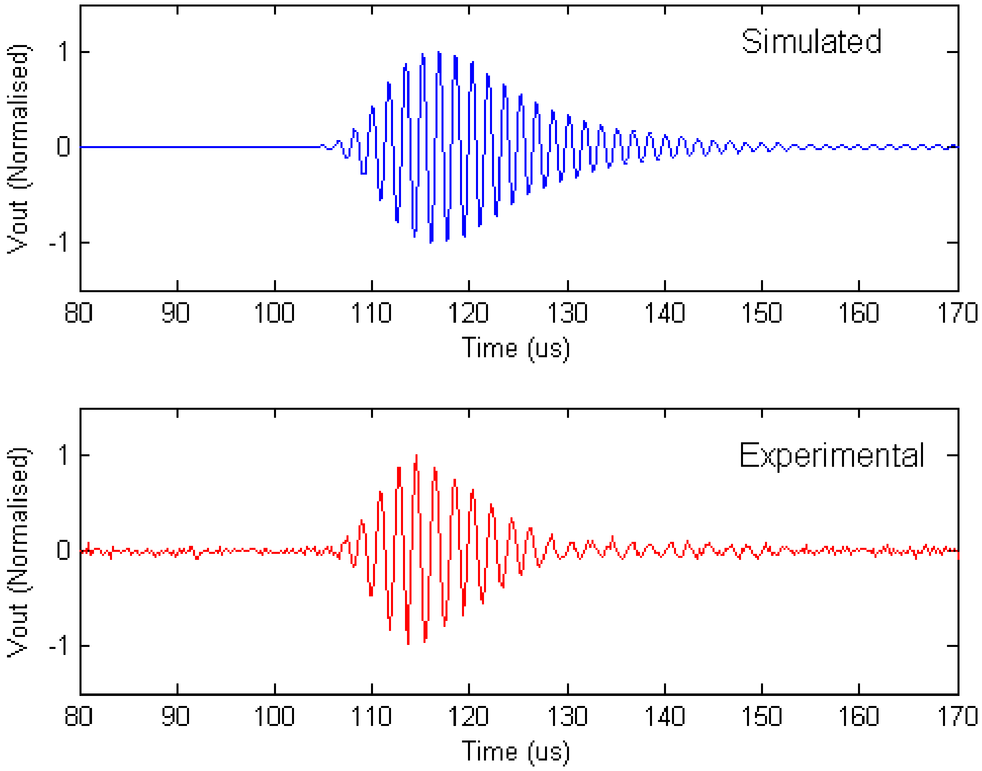


Figure 5.15: Time domain comparison between modelled response (upper figure) and experimentally measured response of air-coupled A_0 generation, propagation and reception in 1mm thick aluminium plate

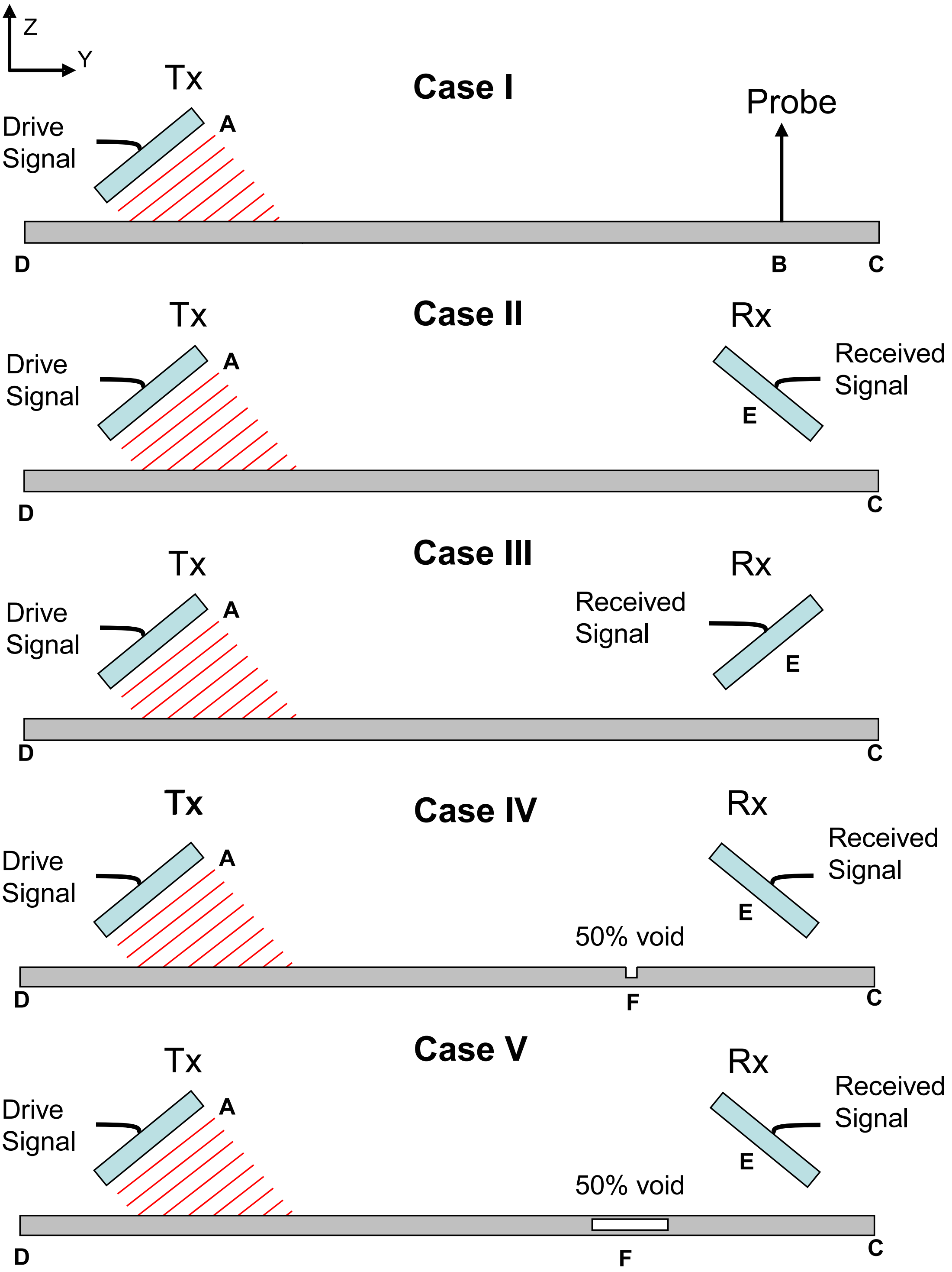


Figure 5.16: Simulation of pitch-catch ray tracing model combined with LISA

wall reflections are visible. In cases IV and V the effect of the defect can be clearly seen, the direct packet is heavily attenuated, there was also a slight delay in the peak caused by some of the energy travelling around the defect. It should be noted that the system was sensitive to voids that are not 100% through the thickness which may be obscured in visual inspection.

Figure 5.18 shows a simulated scan of a 250mm × 250mm 1mm thick aluminium plate with two artificial defects. The transducers had a 10mm diameter and were driven at 600kHz. The defects were 100% through the plate width and were located at (90mm, 170mm) and (150mm, 115mm) with dimensions of 10mm × 10mm and 10mm × 2mm respectively. The second defect was rotated by 30^o relative to the x axis. A single RSA scanned in pulse-echo configuration from (30mm, 30mm) to (220mm, 30mm) in 5mm intervals. This required 39 individual simulations which could be run in parallel over several computers as there was no need for the simulations to be run sequentially. The results are shown as a 2D grayscale plot of the envelope detected receiver time traces. The 'Y' axis was converted from time to distance using the group velocity of the A_0 mode in 1mm aluminium. The result of the scan is shown in Figure 5.19, the image was interpolated linearly by a factor of 5 in the 'X' direction to improve the quality of the image. The 10mm × 10mm defect is clearly visible whereas the 10 × 2mm defect has reduced visibility due to the incident Lamb wave being reflected away from the transducer. The back wall is clearly visible with two shadows cast by the defects.

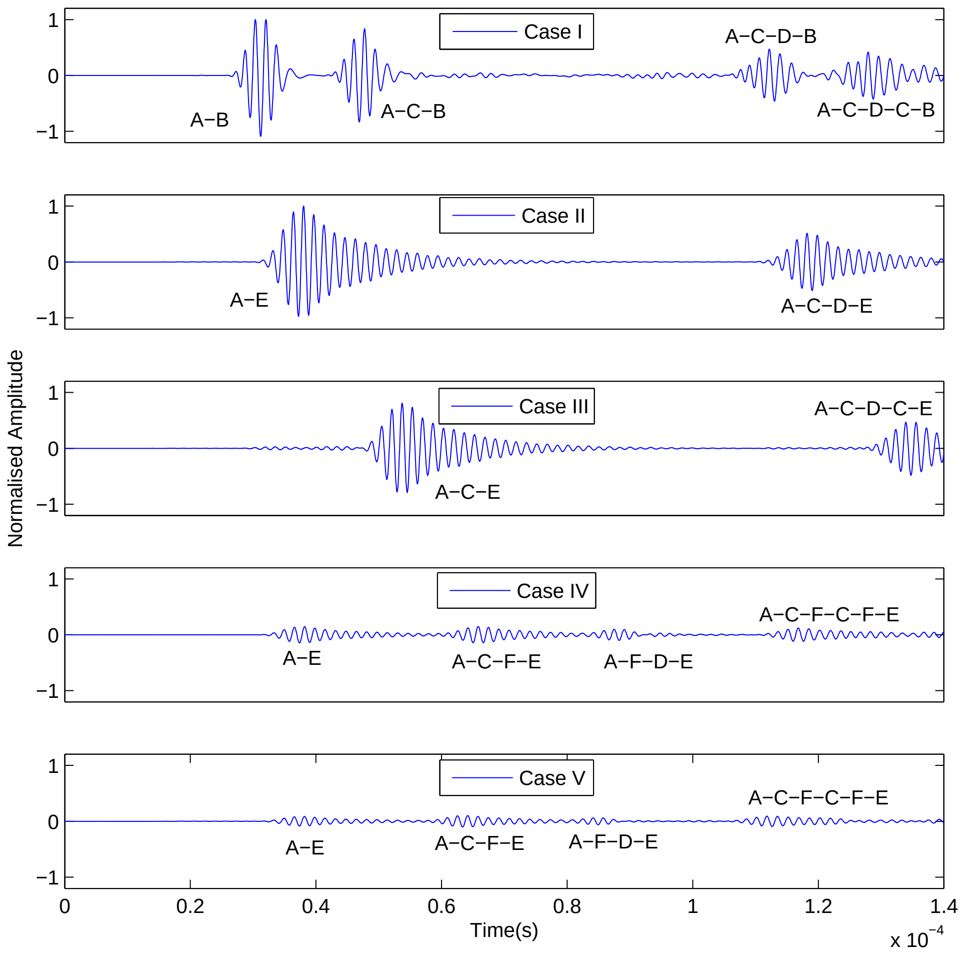


Figure 5.17: Results of simulation of pitch-catch geometry

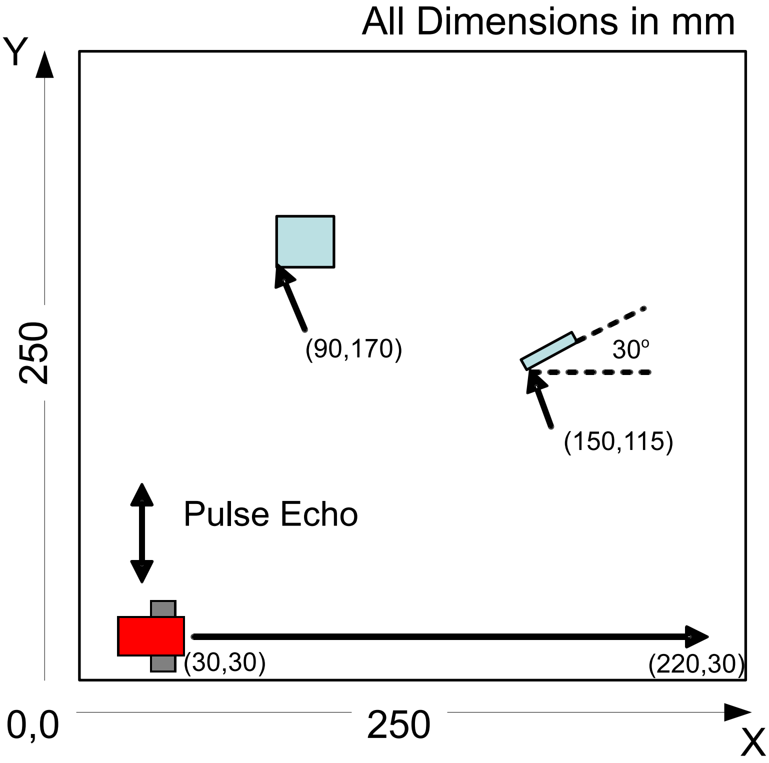


Figure 5.18: Simulation setup of a 3D pulse-echo inspection using a single RSA

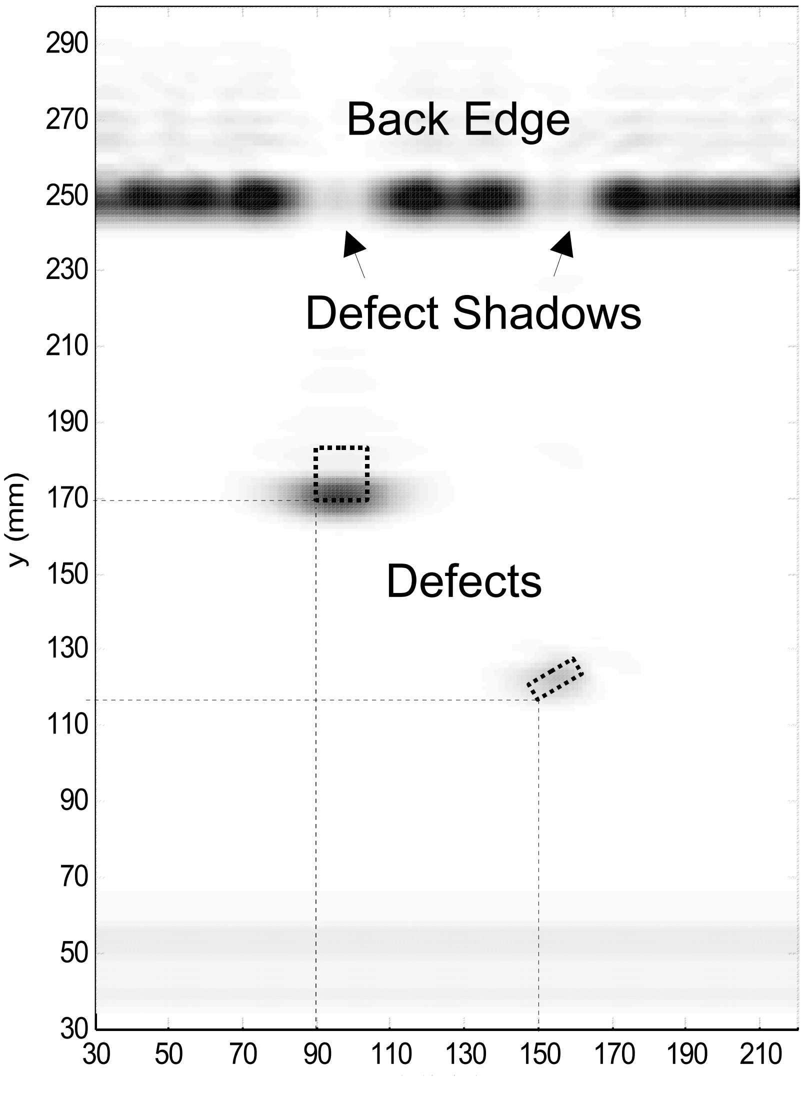


Figure 5.19: Results of a 3D pulse-echo inspection using a single RSA

5.8 Conclusions

A full simulation of an air-coupled ultrasonic scanner for deployment on mobile robotic platforms has been presented and validated against experimental measurements. Generation and reception of ultrasonic Lamb waves were accomplished using air-coupled piezoelectric composite transducers whose response was modelled using a Linear Systems Model [120]. Propagation in sample plates was simulated using the LISA model. This was used instead of conventional FE for two reasons. Firstly to be useful for modelling adaptive inspection strategies implemented by the RSAs, the execution speed of the code is critical. For complex geometries, high calculation times would make it unfeasible to consider optimisation over different propagation paths. Secondly the LISA code could be combined with both the LSM of the transducer Tx/Rx response and the ray tracing approach adopted to couple between the LSM and LISA. The LISA propagation model was extensively validated by comparison of Lamb wave propagation to both a commercial simulation package and a numerical solution to the Rayleigh/Lamb frequency equations.

The complete model was coded in C++ and can be run interactively using a custom GUI running in Windows or in batch mode in either Windows or Linux. The batch code could be configured to run on a High Performance Computer (HPC) allowing multiple simulations to be run in parallel on separate nodes. This is particularly effective for parameter sweeps and for tomographic imaging which requires a series of projections and was used to create Figure 5.19. Since the code is cross platform it can be compiled to run on the RSA's embedded Linux computer and although limited by onboard memory, simple simulations can be run. Future work will look at distributing the simulation over several RSAs or alternatively to implement a system where the RSA can request a simulation over WiFi which is routed over the internet to a computer off-site that actually runs the simulation. Running the simulation on the RSA reduces the power required for communications, but increases power required for onboard computation leading to an optimum depending on the simulation complexity.

Chapter 6

Lamb Wave Imaging Using a Single Robot

This Chapter considers one configuration of robotic ultrasonic inspection. It is the culmination of many threads of investigation and represents one of the work's main achievements. It makes use of the ultrasonic payload developed in Chapter 4, mounted on the robot developed in Chapter 2. The robot is optimally positioned using the Kalman Filter detailed in Chapter 3. The algorithm was developed using the ultrasonic simulation described in Chapter 5. This Chapter shows the first experimental demonstration of an ultrasonic air-coupled inspection robot.

The ultrasonic payload can be configured in one of three modes, these are shown in Figure 6.1. In the local pitch-catch configuration both transducers are mounted on a single robot and examine the area underneath the transducers. The inspection area is approximately equal to the combined surface area of the two transducers. This configuration typically measures signal attenuation. The second configuration is the extended pitch-catch arrangement. In this case, the transducers are split over two robots, one carrying the transmitter, the other the receiver. This extends the coverage to a strip between the robots. In this method, either signal attenuation or time of flight can be measured. Time of flight measurement requires a wireless trigger to activate reception on the receiver robot. In the third configuration the transducers are configured for pulse-echo. The large dynamic range between transmit and receive signals makes it impractical to perform pulse-echo measurements with a single transducer (receiver signals are masked by transmit ringing), so pitch-catch transducers are used to synthesise a pulse-echo transducer. This configuration measures the time it takes for the pulse to travel to and from a reflector. This time is typically converted to a distance using the wave propagation velocity. This Chapter

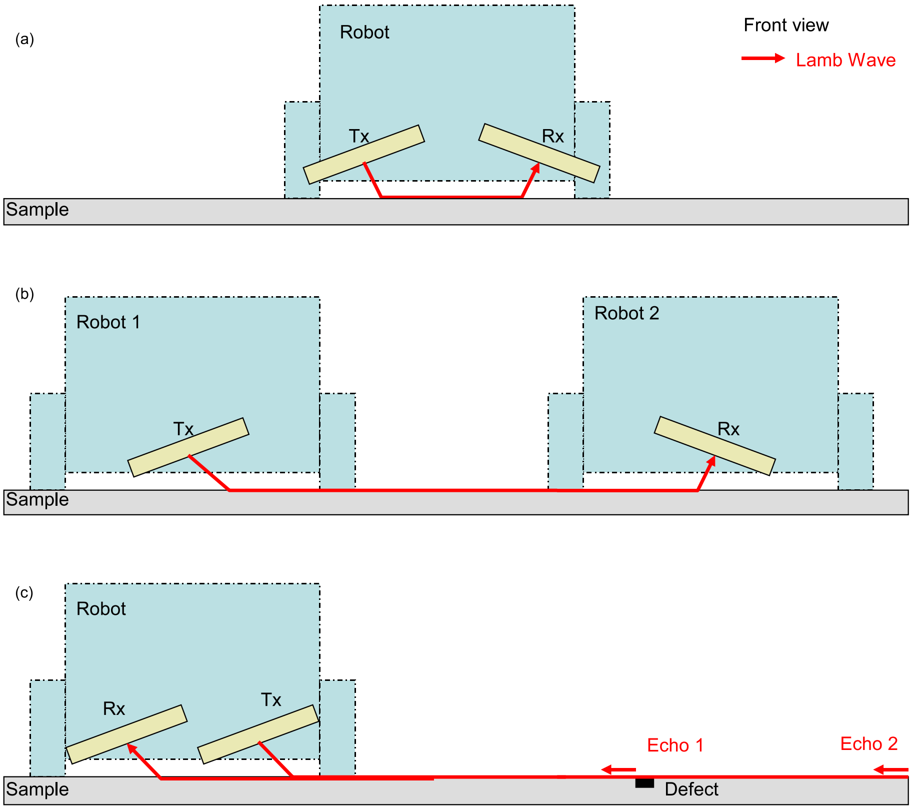


Figure 6.1: Schematic of three possible air-coupled Lamb wave inspection configurations. (a) Local pitch-catch using a single robot. (b) Extended pitch-catch using two robots. (c) Pulse-echo using a single robot.

considers the local pitch-catch configuration, the two remaining configurations are considered in Chapter 7.

6.1 Local Pitch-Catch Configuration

In the local pitch-catch arrangement, the two transducers are positioned side by side on a single robot, as shown in Figure 6.2. They are angled to generate and receive the zeroth order antisymmetric Lamb wave mode (A_0) in the specimen. The angle required for the appropriate mode conversion of the incident planar wave to A_0 can be calculated using Snell's law [94]. This is discussed in more detail in Section 4.4, Chapter 4.

The A_0 Lamb wave mode is dispersive so the phase velocity is dependent on the 'frequency thickness product'. A change in plate thickness and hence frequency thickness product, causes a change in phase velocity. Since the angle for optimum transition is dependant on phase velocity, a change in thickness means that the

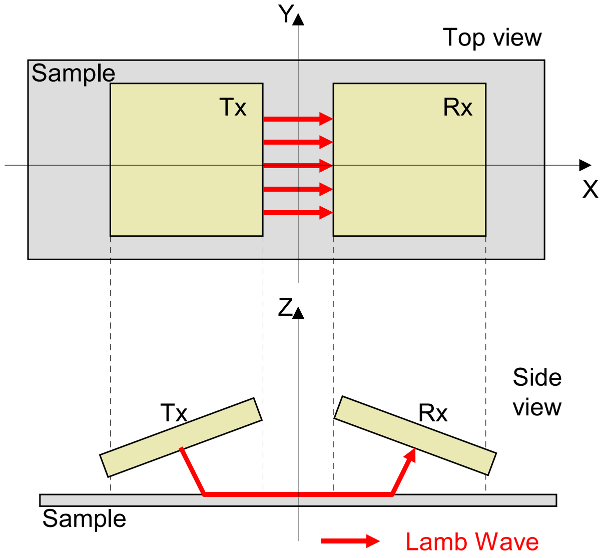


Figure 6.2: Schematic diagram of local pitch-catch experiment showing general layout of the sensors.

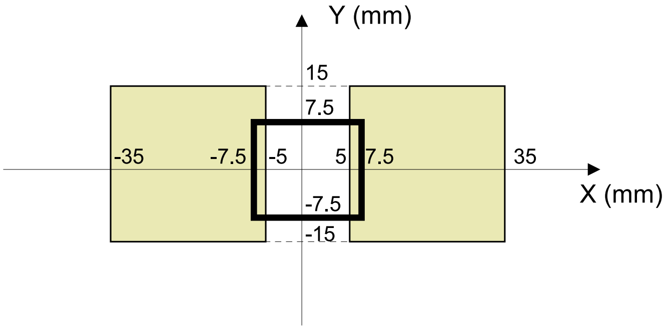


Figure 6.3: Schematic diagram of local pitch-catch experiment showing specific transducer and artificial defect dimensions. The transducers were each 30 x 30 mm separated by 10 mm. The angle relative to the Z axis (Figure 6.2) was only 9.8^o, so the transducers each cover an area of approximately $30 \times 30mm$. The defect was $15 \times 15mm$, 0.5mm deep from the top surface. The sample was 1 mm thick aluminium.)

angle is suboptimal and results in an attenuated signal. It is by this mechanism that defects are detected. Defects may also scatter the Lamb wave further attenuating the received signal. In order to generate the A_0 mode at 600 kHz the angle is set at 9.8^o for 1mm thick aluminium.

A mechanical scanner using this arrangement was demonstrated by Farlow in his PhD thesis [96]. The main advantage of this approach is that the resolution is dependent on the transducer dimensions rather than the excitation signal, so good resolution is possible even at low frequencies. This arrangement is sensitive to defects at any point thorough the material thickness making it attractive over other techniques, such as visual inspection, that can only detect surface breaking defects. The main limitation with this approach is that in order to scan the entire surface with a fine resolution, the RSA must cover every point on the surface leading to very long scan paths.

6.2 Payload Sensitivity to Defects

This section briefly considers the sensitivity of the Lamb wave payload to an artificial defect. It should not be compared with the general study into Lamb wave interaction with defects which takes a detailed took at how the Lamb waves interact with defects such as cracks, corrosion pits etc. The results detailed herein are illustrative in nature.

The ultrasonic simulation described in Chapter 5 was used to simulate the response to a 50% depth, 15 × 15 mm defect in a 1 mm aluminium plate sample. The payload was configured as shown in Figures 6.2 and 6.3. The transducers were 600kHz, 30 × 30 mm, held 10 mm apart at the appropriate incident angle to generate the A_0 Lamb wave mode. The transducers were excited with a 10 cycle tone burst. The defect was positioned at 0,0 mm. An area of 110 × 100 mm was scanned, this was divided into 5 × 5 mm cells, and the payload was positioned at the corner of each cell. This required 440 individual simulations. The time history for each receiver was captured, enveloped using a Hilbert transform and the maximum amplitude extracted.

Figure 6.4 plots the peak receiver amplitude for the payload positioned relative to the position of the defect. The defect is shown in black. The receiver amplitude is mapped to a colour scale going from blue to red. Red areas represent sections where the Lamb wave has been attenuated by the defect. The normalised receiver amplitude is plotted in 3D in Figure 6.5. When the defect is directly under the transducer there is severe attenuation of the receiver signal. The magnitude of the attenuation is reduced as the defect moves out from under the transducer. Each measurement inspects an area of approximately 70 × 30 mm, leading to the overestimation of defect size visible in Figure 6.4.

The receiver output is attenuated when either the transmitter or receiver pass over a defect (spatial convolution of sensor and defect). Therefore, the final image consists of two identical images that are superimposed, but shifted by an amount equivalent to the transducer separation. Farlow removed the double image by multiplying together pixels that were separated by the transducer separation. This enhanced features that were evident to both the transmitter and receiver, whilst removing noise that was only evident in one or the other. An example of this processing applied to Figure 6.4 is shown in Figure 6.6.

The simulation was verified experimentally by R. Summan [139], using the ultrasonic payload mounted on an X-Y scanner. The experimental setup was the same as

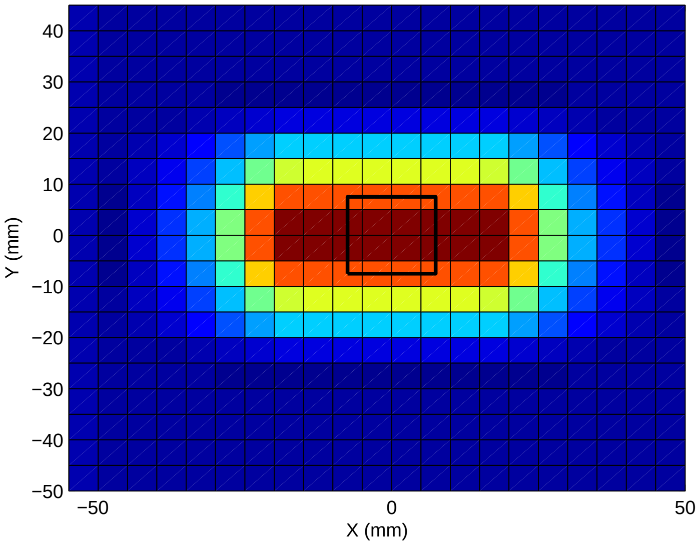


Figure 6.4: The simulation of an Lamb wave raster scan. The figure plots the receiver amplitude at 440 locations over the surface of the sample. The defect is shown in black. The receiver amplitudes have been normalised relative to the defect free region of the plate.

in the simulation. The result is shown in Figure 6.7. The experimental results agree reasonably well with the simulation, both showing a trough in receiver amplitude ranging from X = -40 to 40 mm and Y ranging from -20 to 20 mm. Simulation predicts a 78% reduction in receiver amplitude directly over the defect. The experimental results show this to be even greater at 95%. Since the defect is not large enough to completely obscure the transducer, it is probable that the difference in minimum amplitude is due to noise in the experimental results. This noise is so severe in the experimental results, that it actually hides the defect in some locations, such as X = -25mm, Y - 5mm. The raw signal at this location is indistinguishable from the signal of a defect free region, suggesting that a Lamb wave has been transmitted. The most likely cause of this is transducer misalignment. If the transmitting transducer is not exactly positioned at the required incident angle, the Lamb wave 'packet' does not construct perfectly and some of the energy is transmitted backward, rather than forward (i.e. from right to left in Figure 6.1a). The unintended wave travelling backward can reflect on a defect and then constructively interfere with the wave travelling forward. In such cases, it is possible for the defect to amplify the receiver amplitude in some measurement locations.

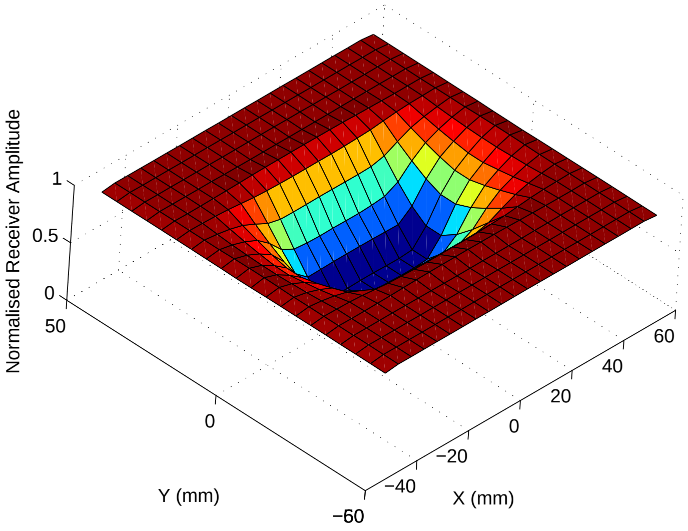


Figure 6.5: The simulation of an Lamb wave raster scan. The 3D figure shows the receiver amplitude at 440 locations over the surface of the sample.

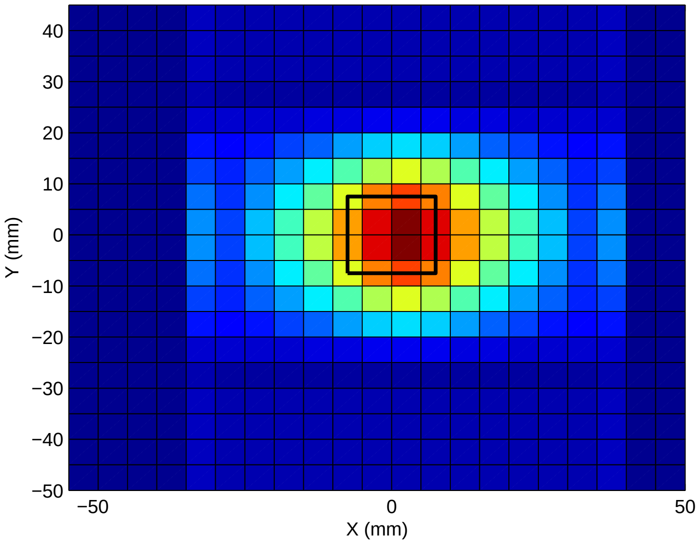


Figure 6.6: This figure shows the application of a post processing algorithm to remove the double. The actual defect is shown in black. The receiver amplitudes have been normalised relative to the defect free region of the plate.

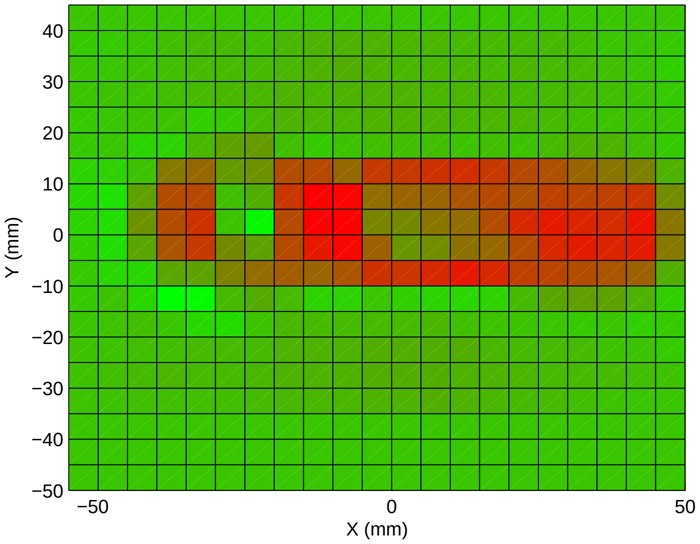


Figure 6.7: This figure plots the experimentally measured receiver amplitude at numerous points over the surface of a 1mm aluminium sample. The receiver amplitudes have been normalised relative to the defect free region of the plate. The colour-map used in the previous plots did not clearly show these results (due to increased noise), so was simplified to a green-red scale.

6.3 Local-Pitch Catch Imaging

Work that was closely related to this thesis, but led by Summan [139], built upon Farlow's work [96], by integrating a probability of detection model into the Lamb wave scanner. Several defect free plates were raster scanned. For each plate a probability density function was created that plotted likelihood of receiving a change in received Lamb wave signal at any location on the plate. This was practically created by dividing the signal amplitude range into numerous band, and plotting the number of occurrences of each band (histogram), irrespective of payload position. The results showed a Gaussian around the mean received amplitude. This is illustrated in Figure 6.8. An amplitude threshold was then chosen at a -3σ deviation from the mean receiver amplitude for the defect free case. Deviations beyond this threshold were taken to be indicative of a defect.

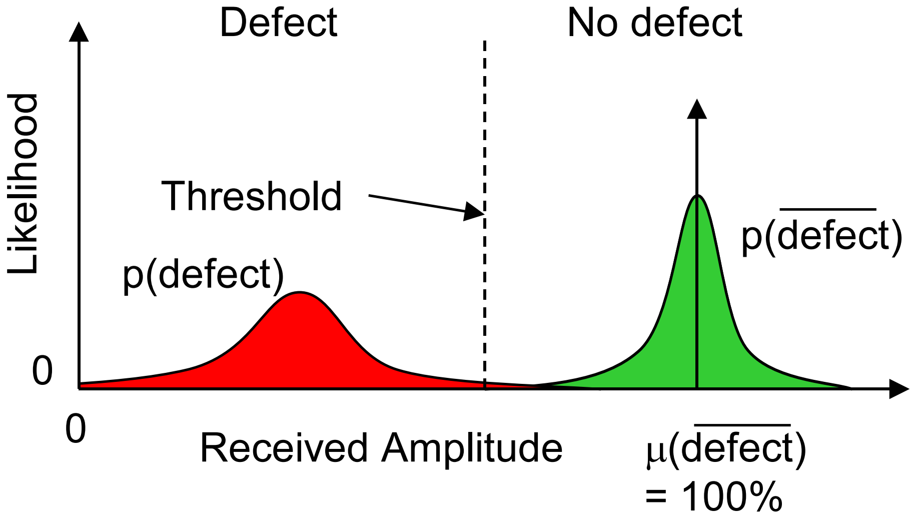


Figure 6.8: The probability density function for air-coupled Lamb wave Payload over regular and defective regions

When this approach was integrated into the miniature robotic vehicle platform, inconsistent transducer alignment presented itself as a baseline shift in receiver amplitude. This was most prevalent if the sample was not perfectly flat, or the robot tyres perfectly uniform. As discussed in Section 4.5.3, Chapter 4, a 0.66^o missalignment in one direction causes a 50% drop in received amplitude. Since the missalignment can occur in both X and Y axes, the tolerance is extremely tight. Unfortunately, the defect detection model was extremely sensitive to a receiver amplitude baseline shift. Consider Figure 6.9. The top diagram shows the ideal case; the defect is clearly identified by a dip in receiver amplitude. In the lower diagram there is an inconsistent amplitude baseline, caused by a slightly uneven plate sample. In this case it is impossible to set a satisfactory threshold. Threshold 1 is too high, incorrectly classifying most of the plate as defective. If the threshold is lowered to below the minimum baseline, the system becomes insensitive to all but the most extreme defects. This made the technique impractical to implement experimentally.

An alternative approach is to map the amplitude of the transmitted Lamb wave. This obviates the need of a threshold and presents defects as sharp discontinuities in the image. This approach is implemented in the next section.

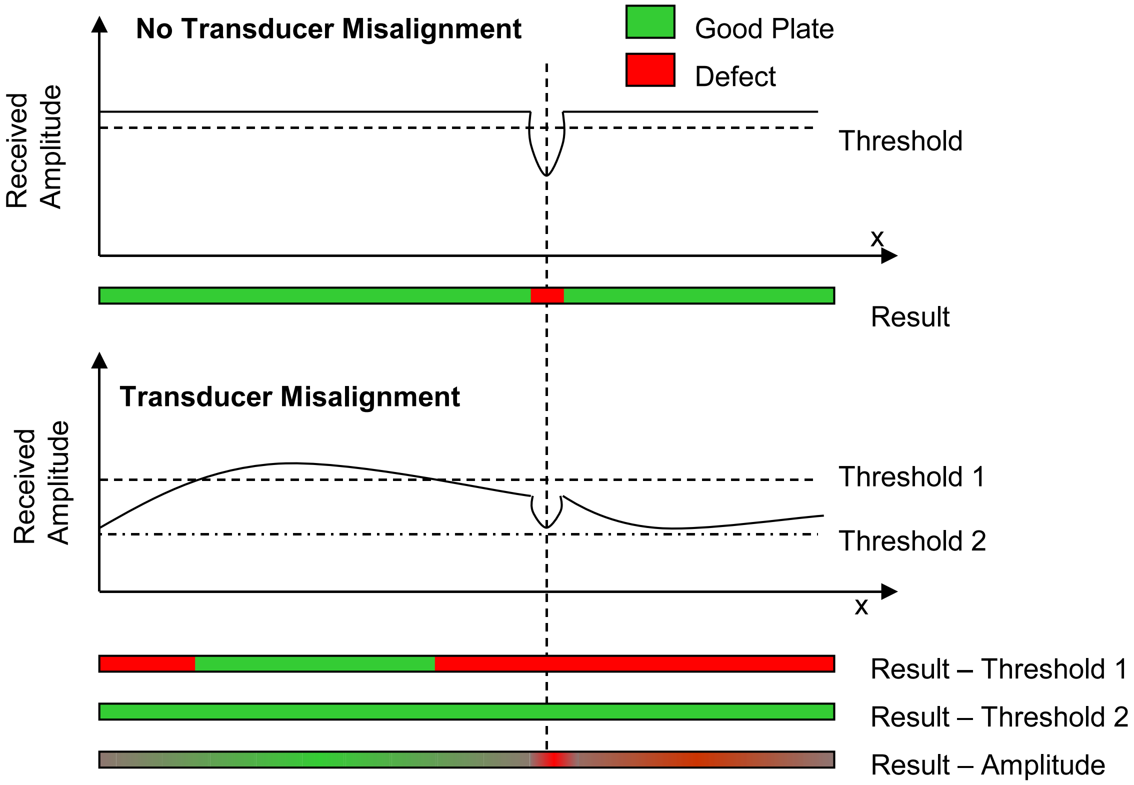


Figure 6.9: The effect of misalignment of transducers on thresholding. The top diagram shows perfect alignment where thresholding works well. In the bottom diagram, small changes in the alignment result in poor defect detection

6.4 The Implementation of Lamb Wave Scanning

In his thesis, Farlow [140] took the measurement location as the centre of the two transducers. This is the origin in Figure 6.2. In reality the measurement is spread over the transducer active area, which is 30 x 70 mm for the payload described in Chapter 4.

The approach taken in this work was to divide the sample into a fine mesh of cells and to apply new measurements to all relevant cells. This is shown in Figure 6.10 A. The approach obviates the need to sample in a regular grid, something that is straightforward when using an X-Y scanner, but more difficult when using the miniature robotic vehicles. Mapping the receiver amplitude removes the need to set a threshold. Missalignment presents itself as a baseline shift that can be filtered out. The important limitation is that the sensor is unable to detect gradual changes in plate thickness, which are mistaken for poor alignment.

In this approach, the same cell is often measured multiple times. The effect is the case in Figure 6.10 sections B and C. In B, each row is sampled twice, once in the top half of the active area and once in the bottom. In this case, the sensible solution would be to take the mean of the two readings. The situation becomes more complicated in C, where several cells are sampled numerous times.

The payload essentially integrates over the active area, so the signal change varies with the amount of the defect under the sensor, typically leading to the trough shape in the received amplitude shown in Figure 6.11. Therefore, the defect has greatest effect on the receiver amplitude when it is positioned near the centre of the sensing area. The same trough shape is visible in Figure 6.5.

Instead of taking the mean of several measurements it would be preferable to weight them according to their position in the sensing area, favouring those in the centre. This is further supported by the physics of angled Lamb wave generation that produces a Lamb wave with high amplitude in the centre, tailing off toward the edges. The trough shape was approximated as a 2D Gaussian function which can be represented using Equation 6.1:

$$P(x,y) = \frac{1}{2\pi\sigma_{sen_x}\sigma_{sen_y}} e^{-\frac{1}{2}\left(\frac{x^2}{{\sigma_{sen_x}}^2}+\frac{y^2}{{\sigma_{sen_y}}^2}\right)} \tag{6.1}$$

x and y are defined relative to sensor's origin (Figure 6.2), σ_{sen_x} and σ_{sen_y} are used to shape the Gaussian and were chosen to approximate the 2D shape in Figure 6.5.

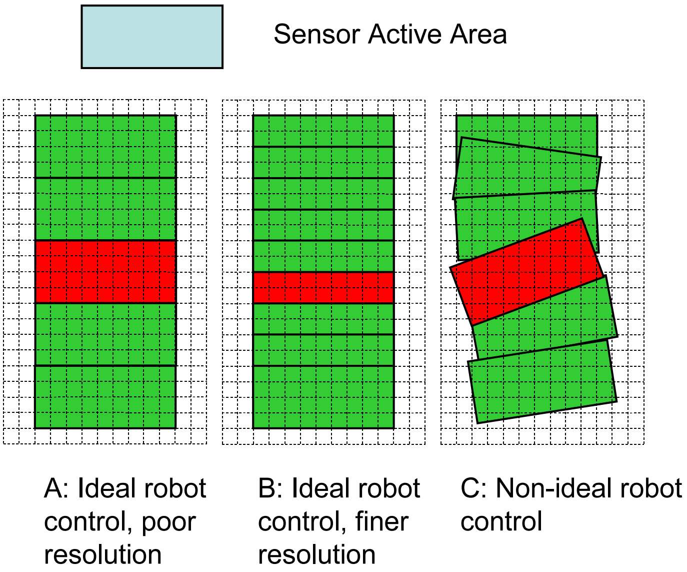

Sensor Active Area
A: Ideal robot control, poor resolution
B: Ideal robot control, finer resolution
C: Non-ideal robot control

Figure 6.10: Examples of overlapping measurement regions

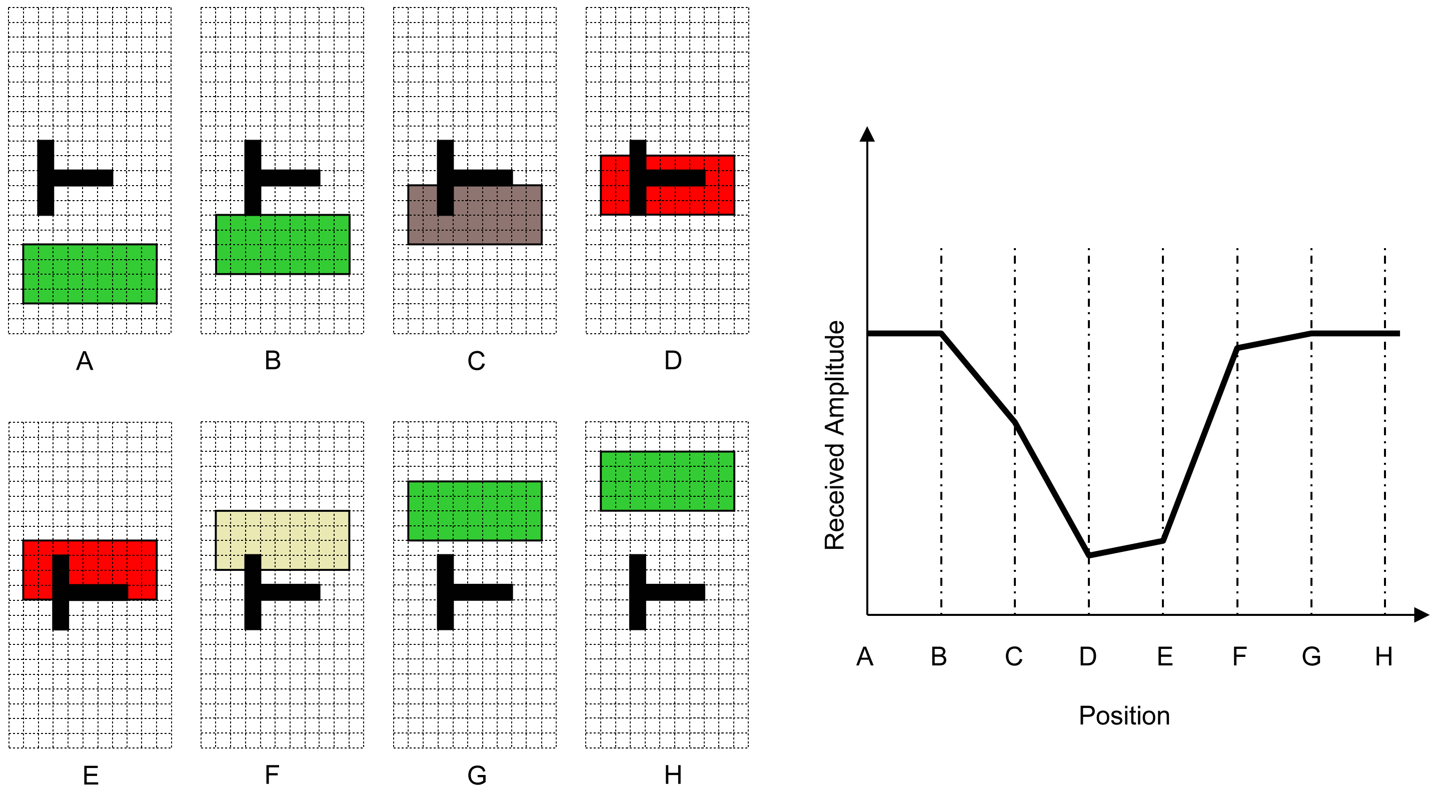


Figure 6.11: Graphical representation of a Lamb wave pitch-catch sensor passing over an artificial defect. Colour is used to give an illustrative representation of signal amplitude.

Each cell in the sample mesh retains a history of each measurement/weighting pair for every measurement that covers the cell. The final cell value is calculated as a weighted average of all amplitudes.

The simulation in Section 6.2 was reprocessed using the new algorithm. Figure 6.12 shows the resulting image. The image fidelity is far worse than the results generated in Section 6.2. This is due to the measurement being applied to the whole measurement area. As the 70 × 30 mm active area is scanned horizontally across the defect, the defect measurement is applied to cells over an area of 140 × 30 mm. The horizontal resolution can be improved by including results from an additional scan with the payload rotated by 90^o. This is shown in Figure 6.13. Figure 6.14 shows the 6dB contour of Figure 6.13.

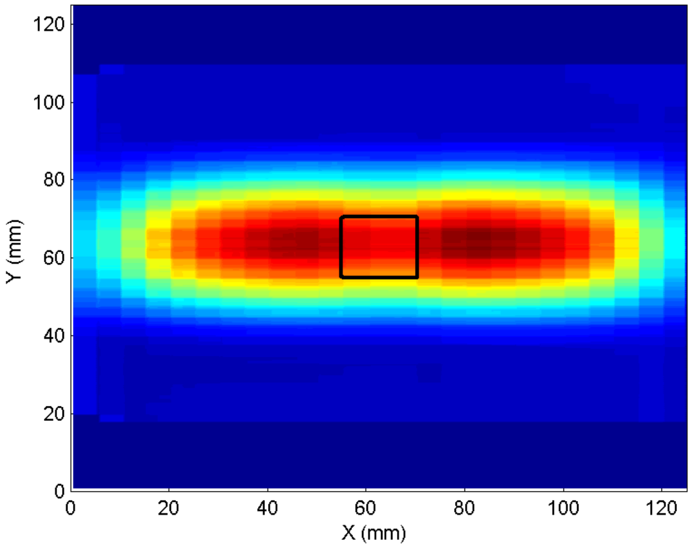


Figure 6.12: The data from Section 6.2 was reprocessed using the new algorithm.

Interestingly, this algorithm can be simplified back to the original algorithm by making the sensor model artificially small, for example a disc 5 cells wide. The result is shown in Figure 6.15. This is equivalent to the image in Figure 6.4. However, unlike the original algorithm, there is no need to sample in a regular grid. Measurements can be applied in any position, with the payload in any orientation.

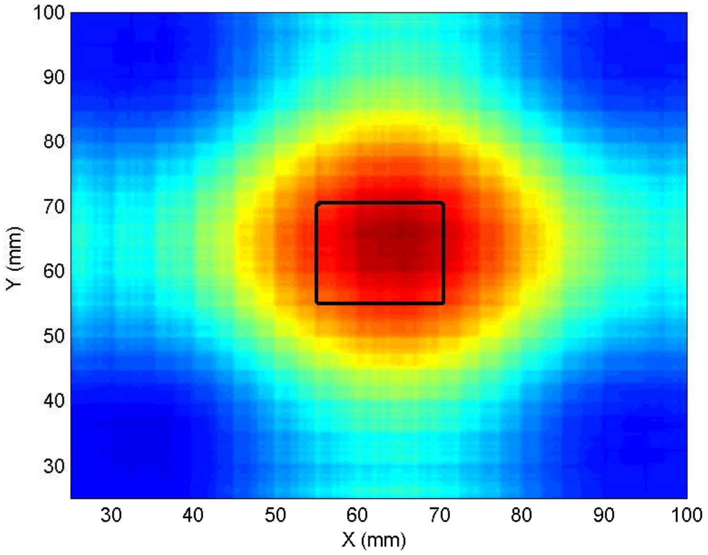


Figure 6.13: The scan in Section 6.2 was repeated with the payload rotated by 90^o. This data was combined with the data from Figure 6.12 in order to produce an image with higher fidelity.

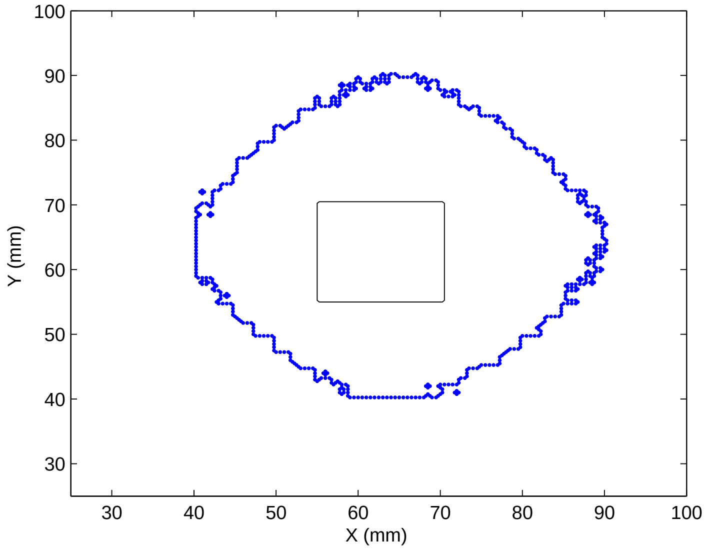


Figure 6.14: A 6dB contour was applied to Figure 6.13

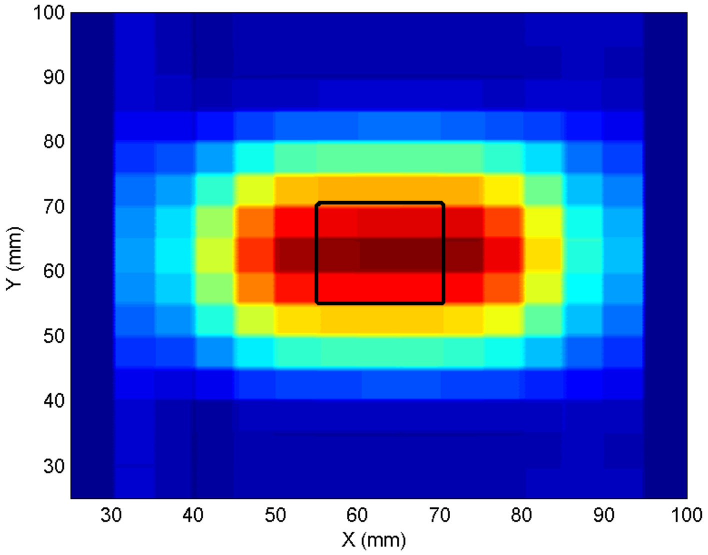


Figure 6.15: Simplifying the sensor model to a small disc, makes the algorithm equivalent to the approach taken in Section 6.2. This Figure is comparable to Figure 6.4.

The fidelity of the image can be improved by modifying the algorithm to give precedence to measurements suggesting a cell is defect free. A high amplitude signal is only possible if there is no defect under the transducer, in such a case all cells under the transducer must be defect free. Consider the top row of cells measured in Figure 6.11 B. This row was measured again as the third row in measurement C. Instead of taking the weighted average of the two measurements, it is preferable to take measurement B, over measurement C, for this particular row. Figure 6.16 shows the application of this new algorithm to the same data. For reference, this algorithm will be refereed to as the 'cell maximum' algorithm, since it returns the maximum measurement for each cell. A 6dB contour is shown in Figure 6.17, providing an excellent estimation of defect size. In order to ensure robustness, the algorithm only takes the minimum of the measurements with a weighting above a 20% threshold. The cell maximum algorithm is far more sensitive to noise than the weighted average version. In the weighted average version, spurious readings would eventually be 'averaged out', but this is not this case here. The algorithm worked well in simulation, but was intolerant to the high level of noise evident in experimental results. For example the spurious readings observed in Figure 6.7 at X = -30mm, Y = 5mm would cause all cells covered by this measurement to be reported defect free. When implemented, this meant that whole defects were often masked. The cell maximum algorithm was intolerant to the reasonably high level of noise on the Lamb wave defect detection measurement. Improving the SNR of the sensor should resolve this issue.

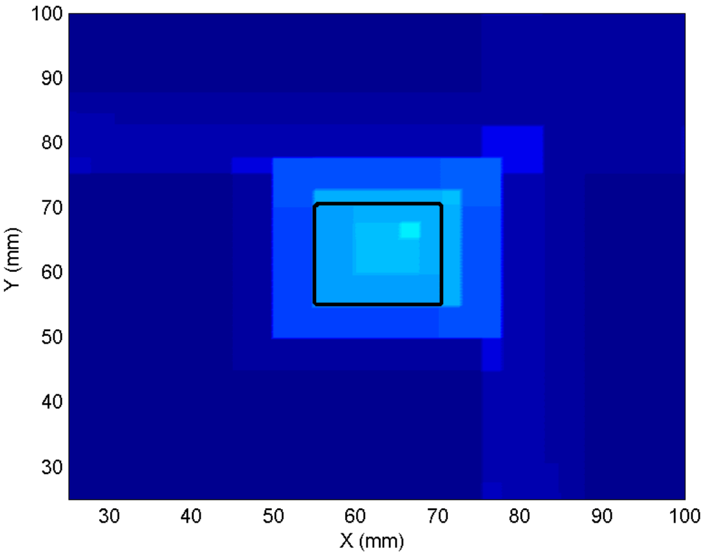


Figure 6.16: The data from Section 6.2 was reprocessed, giving preference to data showing cells to be defect free.

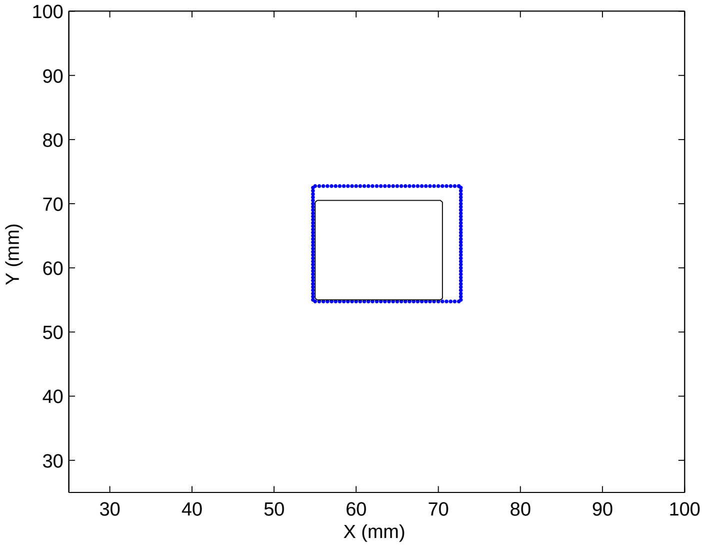


Figure 6.17: A 6dB contour was applied to Figure 6.16

6.5 Integration of Robot Positional Uncertainty

As discussed in Chapter 3 the robot's position contains a degree of uncertainty. This uncertainty is captured by the Kalman filter covariance matrix (Section 3.5.2, Chapter 3). Assuming the states X, Y and θ are mutually exclusive, i.e. all terms in the covariance are negligible except for the diagonal containing the variances, it is straightforward to refine the sensor weighting to account for the robot positional uncertainty. The robot's positional uncertainty is defined by three variances σ_{robot_x}, σ_{robot_y} and σ_{robot_θ} defining uncertainty in X, Y axis and orientation respectively.

The positional uncertainty was integrated in two stages. The first of which was to convolve the positional uncertainty for X and Y with the sensor model, which can conveniently be expressed as another Gaussian function:

$$P(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}\left(\frac{x^2}{{\sigma_x}^2}+\frac{y^2}{{\sigma_y}^2}\right)} \tag{6.2}$$

where:

$$\sigma_x = \sigma_{sen_x} + \sigma_{robot_x}, \qquad \sigma_y = \sigma_{sen_y} + \sigma_{robot_y}$$

The second stage was to integrate the uncertainty in the robot heading, θ. Since the sensor module is mounted in front of the robot, the transform from the robot coordinate system to the sensor coordinate system is simply a +75mm translation along the Y axis. When the robot turns on the spot, the sensor payload follows a circular path of 75mm radius.

Figure 6.18 shows the effect of angular uncertainty on cell weightings. The total sum of the individual weights remains constant, but as angular uncertainty increases the weighting is spread further along the arc. This effect can be thought of as the summation of a series of the 2D Gaussian functions calculated by Equation 6.2 at a range of angles, weighted by the angular uncertainty, σ_{robot_θ}. This is shown graphically in Figure 6.19. It can be represented mathematically as follows:

$$P(x,y) = \int_{\theta=-\pi}^{\theta=\pi} \frac{1}{\sigma_{robot_\theta}\sqrt{2\pi}} e^{\frac{-1}{2}\frac{\theta^2}{{\sigma_{robot_\theta}}^2}} P(x',y')d\theta \tag{6.3}$$

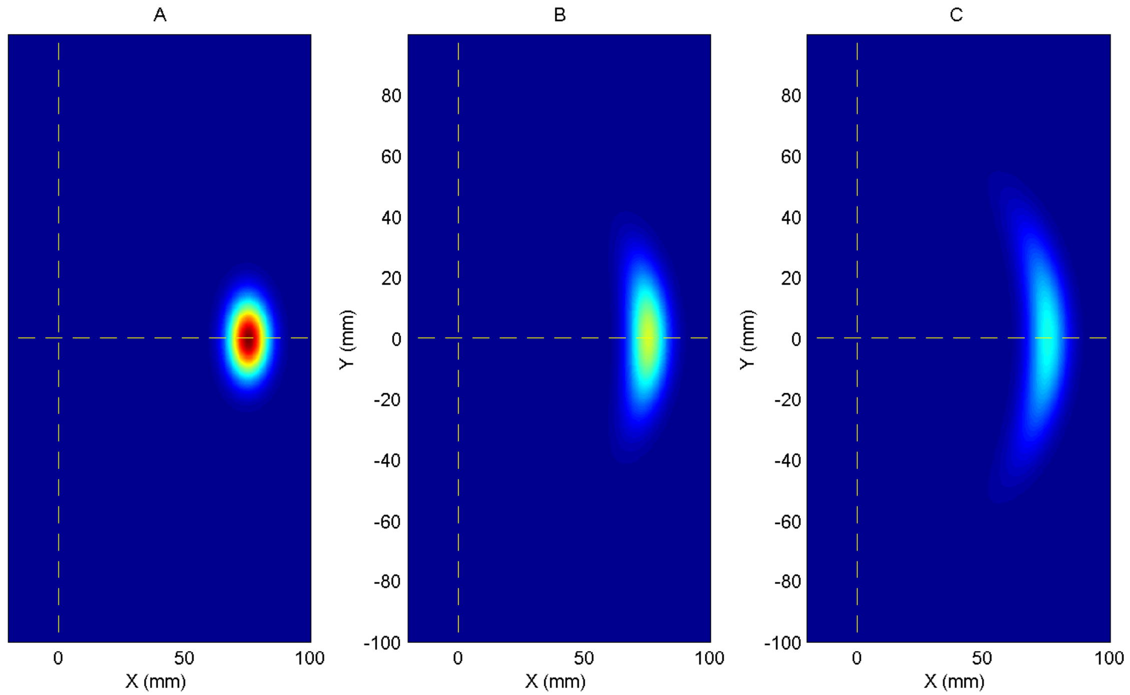


Figure 6.18: Diagram that shows the skewing of the Gaussian probability density function when angular uncertainty is considered. In ‘A’ there is no angular uncertainty, in ‘B’ the angular uncertainty has a variance of 2^o and in ‘C’ the angular uncertainty has a variance of 5^o.

Where:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ y_0 \end{bmatrix} \tag{6.4}$$

and P(x',y') is calculated from Equation 6.2, using x' and y' from Equation 6.4.

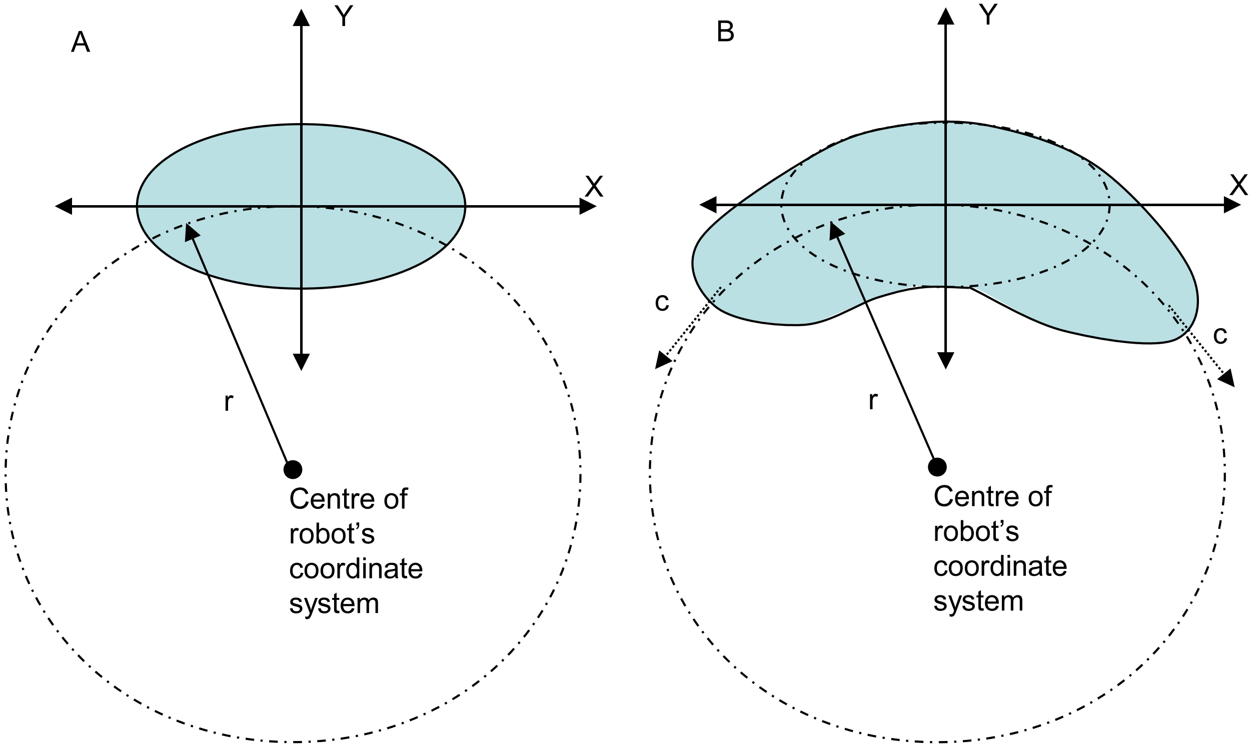

A
Y
X
r
Centre of robot's coordinate system
B
Y
X
c
c
r
Centre of robot's coordinate system

Figure 6.19: This figure shows the effect of angular uncertainty on the measurement probability density function. In A there is no angular uncertainty. The small angular uncertainty in B, skew's the Gaussian function.

6.6 Integration into the Hardware Platform

Each cell in the sample grid retains a history of each measurement/weighting pair. The spatial 'weighting' distribution is calculated for each ultrasonic measurement using the Kalman Filter's estimate of positional uncertainty at the last filter update. The resulting cell amplitudes are calculated as a weighted average of all measurements. In practice, measurements with negligibly low weighting are discarded to save memory (the weighting is stored as a single byte in memory, so weighting below 1/255 are discarded).

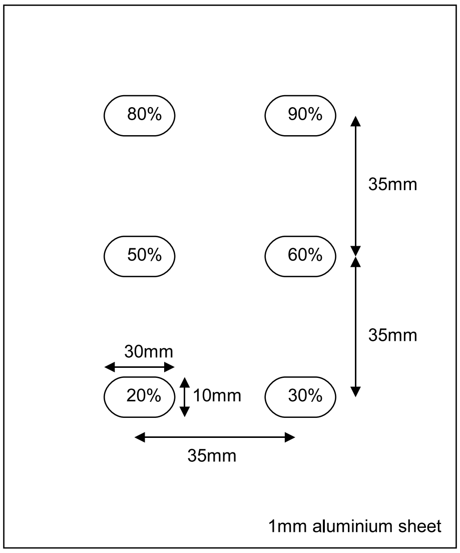
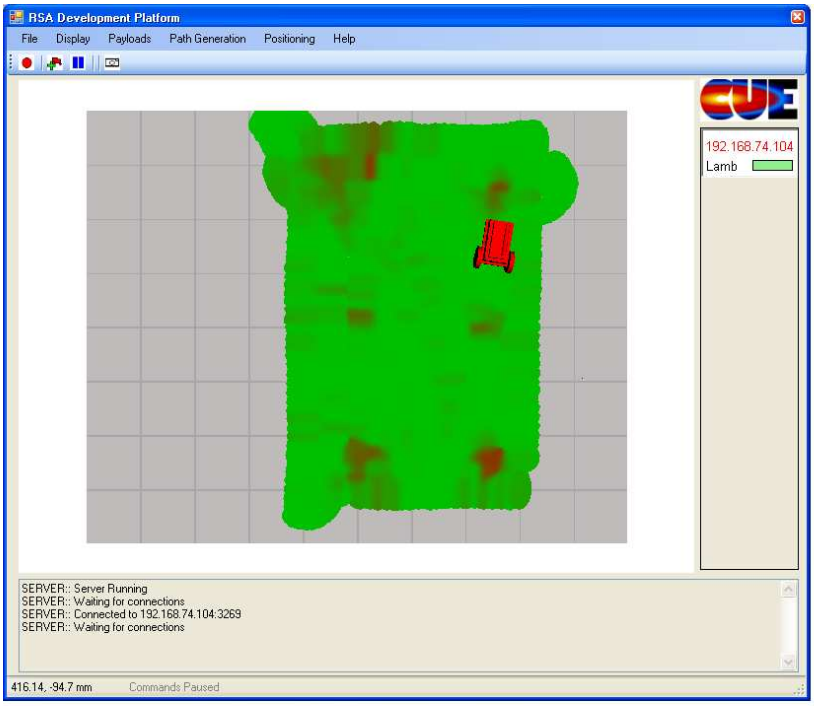


Figure 6.20: Schematic of sample for local-pitch catch experimental scan, containing 6 30×10 mm defects. The defect depth is shown as a percentage of material depth.

The robot described in Chapter 2 was equipped with the payload described in Chapter 4. The weighted average algorithm was integrated into the robots Graphical User Interface. The robot was instructed to scan a 1mm aluminium plate with six artificial defects, each 10 × 30 mm in size and of varying depth ranging from 20 to 90 %. The geometry is shown in Figure 6.20. Figure 6.21 shows the resulting image. All six defects are clearly visible in the pitch-catch scan. Figure 6.22 shows the data in 3D, plotting the probability of a cell being a defect on the z axis. A peak in amplitude toward 1 shows increased likelihood of a cell being a defect. Unscanned regions are assigned an amplitude of 0.5, regions that have been scanned and shown as defect free, have an amplitude approaching 0.



```
SERVER:: Server Running
SERVER:: Waiting for connections
SERVER:: Connected to 192.168.74.104:3269
SERVER:: Waiting for connections
```

```
416.14, -94.7 mm        Commands Paused
```

Figure 6.21: Example scan using of 1mm using air-coupled pitch-catch transducers as captured by Robot Software and displayed in context in robot world.

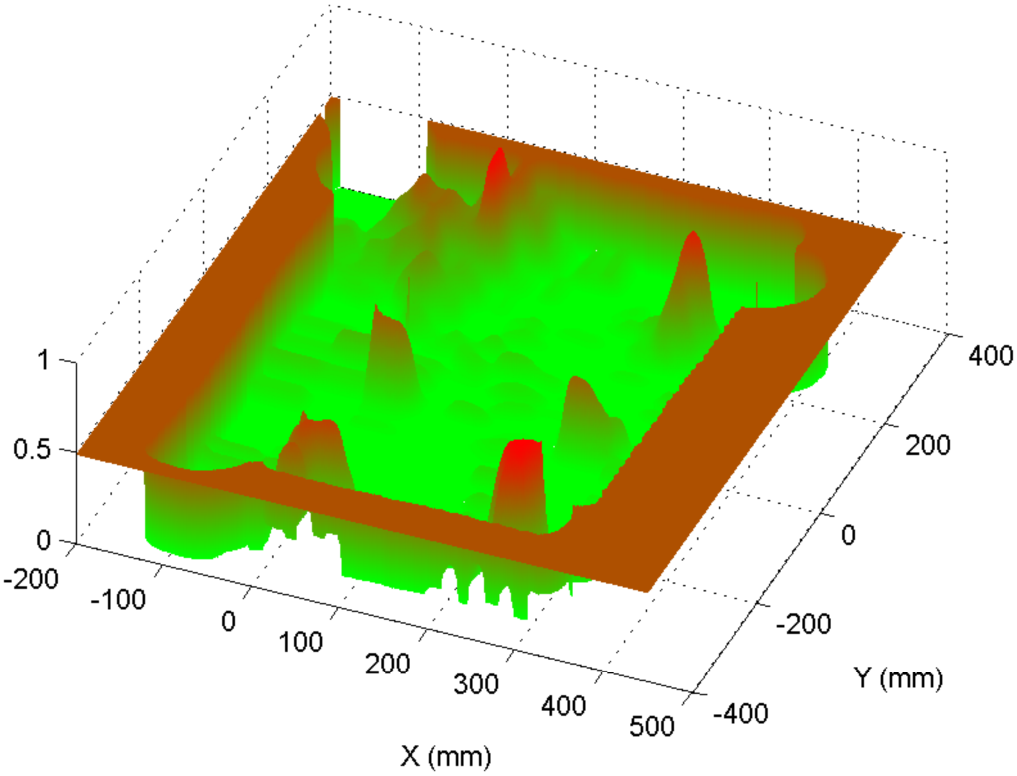


Figure 6.22: Example scan using of 1mm using air-coupled pitch-catch transducers shown in 3D

6.7 Conclusions and Further Work

This chapter has successfully developed a robotic non-contact ultrasonic scanner. The is the first time that a miniature battery operated non-contact ultrasonic scanner has been experimentally realised. The NDE algorithm takes account of robot positional uncertainty in real time, combining results in an optimal fashion, which in itself is a significant step forward in the field of robotic NDE. The algorithm is well suited to ad-hoc results, generated from multiple robots.

In the current form, the algorithm only outputs the probability of a cell representing a defect. If a cell is only scanned once with a very low weighting (right on the edge of a measurement), the result is undistinguished from a cell that has been scanned several times at higher weighting. Future work could exploit this with weighting information to ensure that all cells had been scanned sufficiently. Since all measurement weighting pairs are retained, it may be possible to perform outlier rejection if one measurement is drastically different to the others.

Future work may integrate Farlow's technique to remove the double image [96], into the weighted average algorithm. This could be achieved by generating two sets of results, one based on the transmitter position and the other based on the receiver position. The algorithm output would be generated by multiplying these results together.

The most significant refinement will come from smaller payload transducers, that will provide superior image fidelity.

Chapter 7

Reconfigurable Scanner

Chapter 6 discussed the use of robot mounted non-contact pitch-catch transducers for the inspection of plates. However, the payload had to cover all points on the plate surface, leading to long scan times. An example of this is shown in Figure 7.1 a. Figure 7.1 b shows an alternative approach. It minimises the scan length by propagating Lamb waves over long distances, inspection a large area with each measurement. This arrangement is known as a B-scan. The robot emits a pulse and then listens for echos from defects, or plate boundaries. The location of any reflectors can be estimated from the arrival time of the echos.

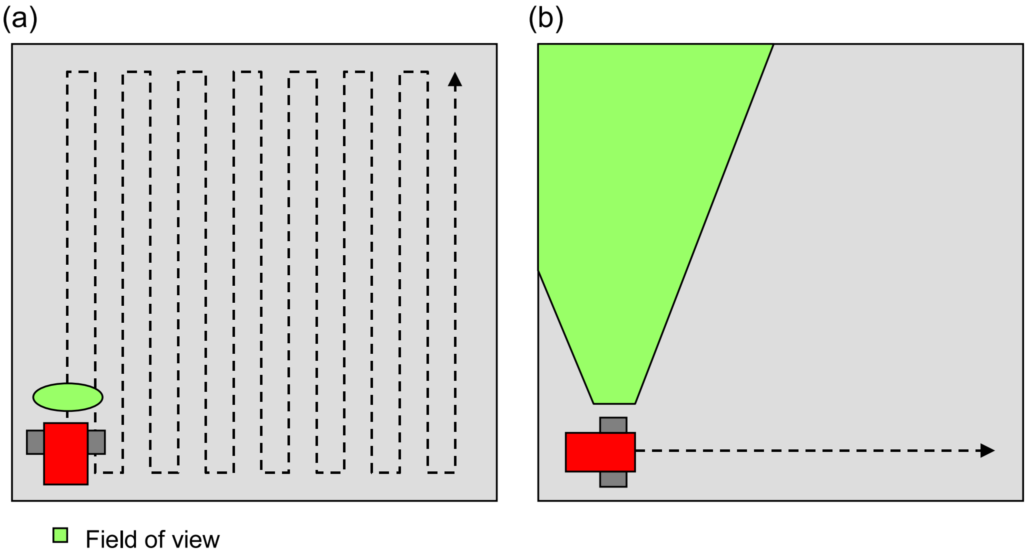


Figure 7.1: Robot scanning strategies
Schematic of two robot scanning modalities. (a) Local-pitch catch, (b) Pulse-echo.

The **S**ynthetic **A**perture **F**ocusing **T**echnique (SAFT) is a well known algorithm for processing B-scan data. This chapter begins by considering SAFT as a means of processing pulse-echo results generated by the robot mounted payload. It uses the ultrasonic wave propagation model discussed in Chapter 5 to simulate different configurations and comments on the applicability of SAFT to the robotic scanner, there were several key findings:

1. In the current set-up the A_0 Lamb wave mode is dispersive. SAFT needs to be modified to accommodate dispersion.

2. SAFT requires a wide transducer beam divergence (field of view) so that each same image point is visible at each transducer location.

3. As the width of the angled piezoelectric transducers increases the beam divergence is reduced, limited the effectiveness of SAFT. The $30mm$ wide transducers created in Chapter 4 have negligible beam divergence. In order to achieve a reasonable beam divergence of, for example 15^o, the transducer dimensions would need to be reduced to 8 × 10 mm. This severely reduces the payload's SNR, making SAFT impractical on the current hardware.

4. In addition to this SAFT's phase compensation is relatively intolerant to robot positional uncertainty.

5. A algorithm was demonstrated that constructs an image from B-Scan data, compensating for dispersion nature of the ultrasonic signals. The algorithm was fairly tolerant of robot positional uncertainty, but did not provided the image fidelity offered by phase compensation (SAFT).

The chapter then moves on to discuss the applicability of other imaging techniques, in particular, tomography [141, 142, 143, 144] and the total focusing method [145, 146]. It found that tomographic algorithms required extensive robot scan paths, making them impractical for robots that need to physically move to each element location (unlike arrays that can be scanned electronically). Tomography algorithms that require ultrasonic Time Of Flight (TOF) data are intolerant or robot positional uncertainty, since inaccurate positioning will change the length of the propagation path, corrupting the TOF measurement.

Finally, the chapter demonstrates a simple algorithm that can be used to position the robot on a particular plate section, giving a degree of contextual awareness. This information can then be used to introduce intelligent scanning strategies. The chapter concludes by stating that considering NDE imaging as a mapping problem would be the most promising research theme to pursue in future work.

7.1 Synthetic Aperture Focusing Technique

The **S**ynthetic **A**perture **F**ocusing **T**echnique (SAFT) [147] is a back propagation technique that enhances pulse-echo B-Scans. It achieves this by correctly superimposing the reflections of a single reflector observed from multiple receivers. Figure 7.2 shows the general principle. Multiple receivers observe the same reflection, but in each case the reflection is time shifted due to a difference in the distances between the reflector and the individual transducer (d_i in Figure 7.2 b). SAFT time-shifts the receiver time histories to compensate for the variable distance. This allows the measurements to reconstruct constructively. SAFT is implemented by scanning the field of view and at each location summing the response from each transducer with the appropriate phase shift. SAFT leads to a theoretical SNR improvement of $\sqrt{N}$ over conventional B-Scans where N is the number of receiver elements [148]. SAFT is usually performed in the frequency domain where it is computationally more efficient.

The frequency domain SAFT algorithm will now be presented [147, 16]:

1. Capture pulse-echo time domain responses from each transducer location, ensuring that Nyquist Sampling is satisfied temporally (sample time $<$ 1/(2 $\times$ highest frequency)) and spatially (transducer spacing $<$ shortest wavelength/2). This yields the 2D array $S(x, y = 0, t)$.

2. Convert the data to the frequency domain by taking the 2D Fourier transform.

$$\bar{S}(k_x, y = 0, \omega) = FFT_{x,t}\left[S(x, y = 0, t)\right] \tag{7.1}$$

3. Apply the back propagator (time delay)

$$\bar{S}(k_x, y, \omega) = \bar{S}(k_x, y = 0, \omega)\exp\left(iy\sqrt{\frac{4\omega^2}{{V_p}^2} - k_x^2}\right) \tag{7.2}$$

The square root term is k_y, rearranged for clarity:

$$k_y = \sqrt{\left(\frac{\omega}{V_p/2}\right)^2 - k_x^2} \tag{7.3}$$

This is simple geometry in $\omega - k$ space, noting that $k = \omega/V_p$. The phase velocity V_p is halved because the transducers are being used in pulse-echo con-

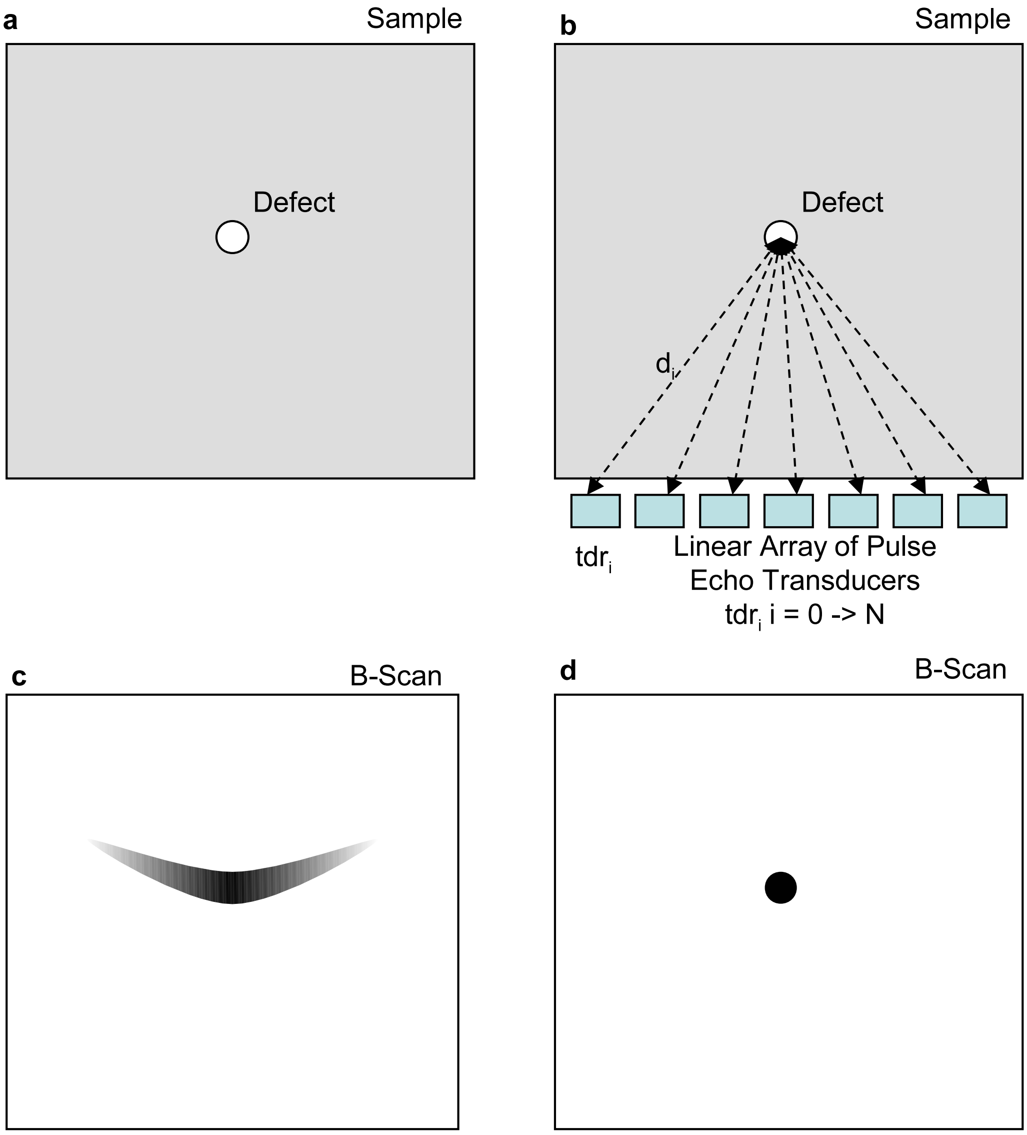


Figure 7.2: Diagram showing the principle of SAFT. a - shows a plate with a single defect. b - shows a linear array of transducers used to inspect the sample, showing the distance between each transducer and the defect. c - shows the resulting B-Scan, due to the defect being visible from several transducers. d - shows the result of SAFT processing to correctly resolve the defect.

figuration, so the energy must travel to the reflector and back. An additional back propagator is applied for the angled wedge [16] (In this case air - see Chapter 4). This takes a similar form to the previous back propagator:

$$P_W(k_y, y, \omega, \beta) = \exp\left(i\left\{(1-\beta)y + \beta D_W\right\}\sqrt{\frac{4\omega^2}{{V_W}^2} - k_x^2}\right) \tag{7.4}$$

Where D_W is the distance the wave must travel in the wedge and V_W is the longitudinal velocity in the wedge. β is a binary operator that is 1 if $y < D_W$. It is used to toggle the expression for points inside and outside the wedge. The complete expression for $\bar{S}(k_x, y, \omega)$ becomes:

$$\bar{S}(k_x, y, \omega) = P_W(k_x, y, \omega, \beta)\bar{S}(k_x, y = 0, \omega)\exp\left(iy\left\{y - D_W\right\}\beta\sqrt{\frac{4\omega^2}{{V_p}^2} - k_x^2}\right) \tag{7.5}$$

D_W can be calculated as follows:

$$D_W = \frac{L_{tdr}}{2}tan(\theta_{tdr}) \tag{7.6}$$

where L_{tdr} is the length of the transducer and θ_{tdr} is the transducers incident angle (For more information, see Chapter 4).

4. Sum over the frequency components:

$$\check{S}(k_x, y) = \sum_{\omega\in\Omega}\bar{S}(k_x, y, \omega) \tag{7.7}$$

5. Finally, take the inverse Fourier transform.

$$B(x, y) = IFT_{k_x}\left[\check{S}(k_x, y)\right] \tag{7.8}$$

SAFT has been adapted for dispersive Lamb waves by making the phase velocity in Equation 7.5 frequency-thickness-product dependent [16, 149]. This is achieved by replacing V_p with $V_p(\omega, d)$, where d is the plate thickness. The plate thickness is assumed to be constant, which is a fairly accurate approximation for small defects in an otherwise uniform plate. The algorithm correctly recreates peaks at the group velocity by summing the effect of each frequency component. This accounts for the effects of dispersion on the received wave.

The LISA simulation tool (Chapter 5) was used to simulate SAFT imaging using air-coupled Lamb wave transducers. The sample chosen was an aluminium plate of dimensions 125mm × 125mm × 1mm. The plate had a single 5mm diameter 50% deep defect (from top surface), centred at 62.5mm × 70 mm. The LISA simulation had a mesh density of 0.5 mm (cubic cells), which gave an A_0 accuracy of 98.4% (See Section 5.4.2, Chapter 5). Ultrasonic propagation was simulated for 85 μs, giving enough time for the Lamb wave to reach the back wall and reflect back to the transducer. The computation time for 85 μs of simulation time was approximately 2 minutes on a quad core 2.6GHz PC with 2GB RAM. The 15 × 10 mm transducer was positioned at 50 positions of 1.5mm spacing, from (x=25mm, y=25 mm) to (x=100mm, y=25mm). This required 50 ultrasonic simulations. The transducer was excited with a 3 cycle tone burst. Figure 7.3 shows the simulated Lamb wave B-Scan. A Hilbert transform has been used to envelope the received signal. The defect is visible at x = 62.5mm, t = 38μs. The horizontal bar at t = 10μs is essentially a reflection of the generated Lamb wave, back into the receiver.

Figure 7.4 shows the SAFT reconstruction of the data in Figure 7.3. The y axis is correctly reconstructed, but there is no improvement in defect resolution. The result can be explained as follows; the 15mm wide air-coupled transducer is very directional; the generated Lamb wave exhibits minimal beam spread, and receivers are only sensitive to waves close to normal incidence. The arc shape in Figure 7.2 is not evident, as only the base of the arc is detected. The next section examines the factors affecting the transducers field of view in order to try and improve the suitability of the transducer to SAFT processing.

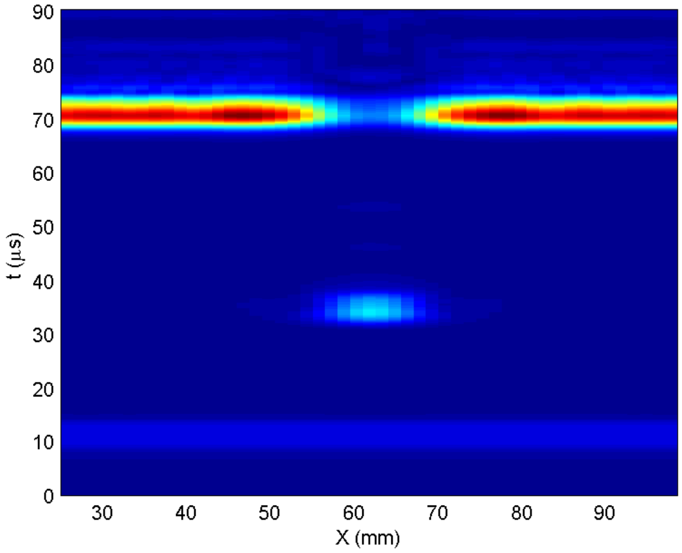


Figure 7.3: B-scan, taking 50 measurements at 1.5mm spacing. The sample is 1mm thick aluminium plate (125mm × 125mm) with a 5mm diameter 50% defect at 62.5mm × 70 mm. The transducers were on a horizontal line from (25,25) mm to (75,25) mm. The transducers are rectangular 15 mm x 10mm held at a 9.8^o angle.

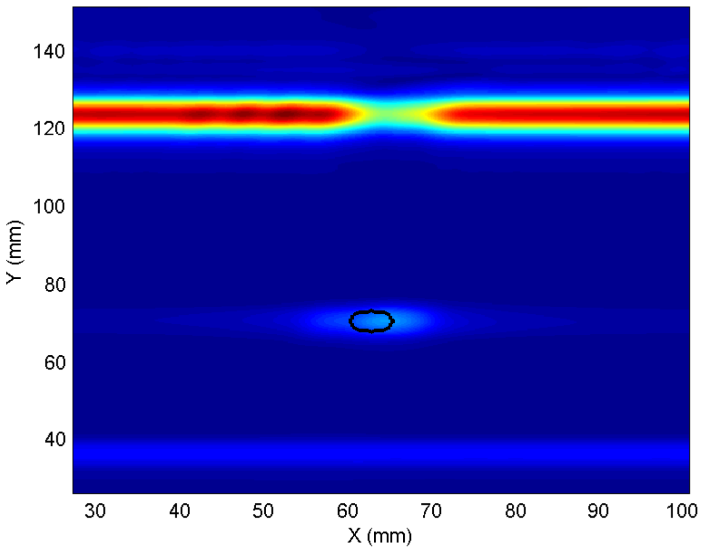


Figure 7.4: SAFT Processing of B-Scan in Figure 7.3. The -6dB defect outline is shown in Figure 7.21

7.2 Transducer Beam Divergence

Reducing a transducer's width increases the beam spread, thus increasing the field of view. The LISA simulation tool was used to characterise the effects of transducer width on beam spread. The simulation consisted of a 125 x 125 x 1mm aluminium plate. The transducer was positioned at (62.5, 25mm) at the incident angle required to generate the A_0 Lamb wave mode(9.8^o). The mesh density was again 0.5 mm^3. Figure 7.8 shows the beam profile of three different transducer widths; 3mm, 9mm and 15mm. The rectangular transducers were all 10mm long and assumed to be ideal planer sources at 600kHz. The transducers were again excited with a three cycle tone burst. The Lamb wave beam profile was created by recording the maximum observed out-of-plane displacement at each sample point on the plate surface. Figure 7.5 shows the profile for a cord placed 62.5mm in front of the transducer (y = 87.5mm). Figure 7.6 shows the peak amplitude from Figure 7.5 plotted against width.

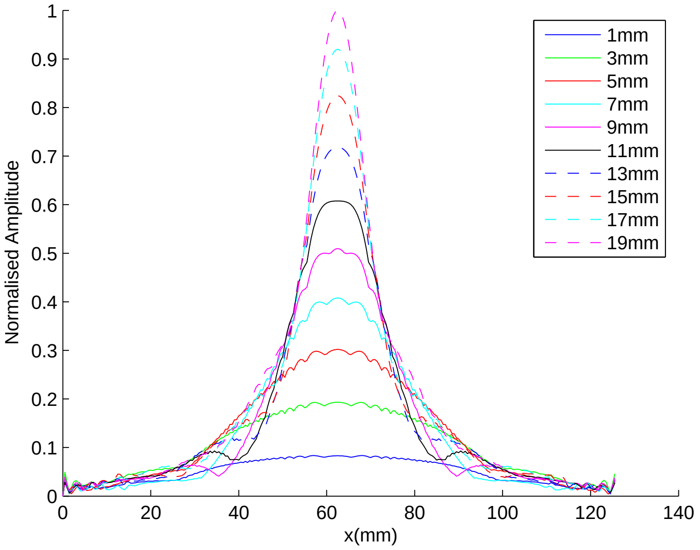


Figure 7.5: Beam profile of a horizontal cord positioned 62.5 mm in front of transducer (y = 87.5 mm) for a range of transducer widths.

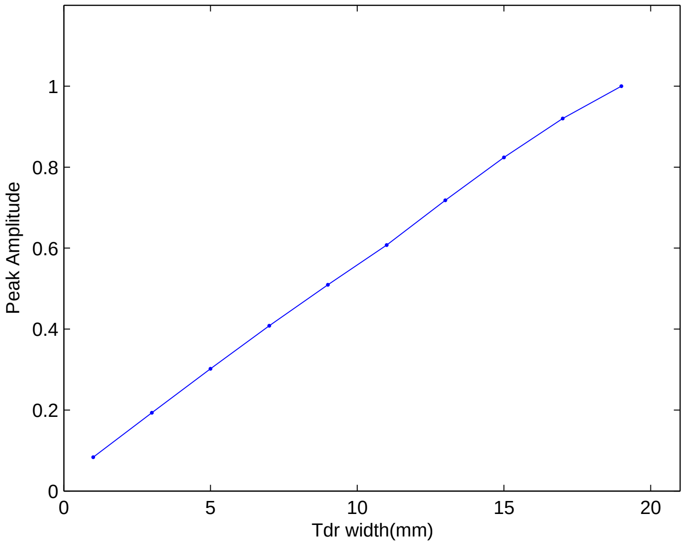


Figure 7.6: Peak amplitude from Figure 7.5 plotted against transducer width

Beam divergence θ_b is defined in Figure 7.7 b. An algorithm was written that scanned the Lamb wave beam plot from y = 0 to y = y_{max} and recorded the maximum divergence at -6dB (50%) relative to the maximum amplitude of a horizontal cord at y from x = 0 to x = x_{max}. This provided the dashed lines shown in Figure 7.8. Beam divergence was plotted against transducer width in Figure 7.9. The theoretical -6dB beam spread, θ, from a square aperture of width, W, can be calculated as [91]:

$$\theta = 0.514 \sin^{-1}\left(\frac{\lambda}{W}\right) \tag{7.9}$$

where λ is the wavelength of the propagating wave. The simulation results show good correlation with theory.

The results show that as the transducer width and hence aperture is reduced, the beam divergence increases; at 19mm the divergence is only 6.7^o. Reducing the transducer width from 15mm to 3mm increases the beam divergence from 7.9^o to 26.4^o.

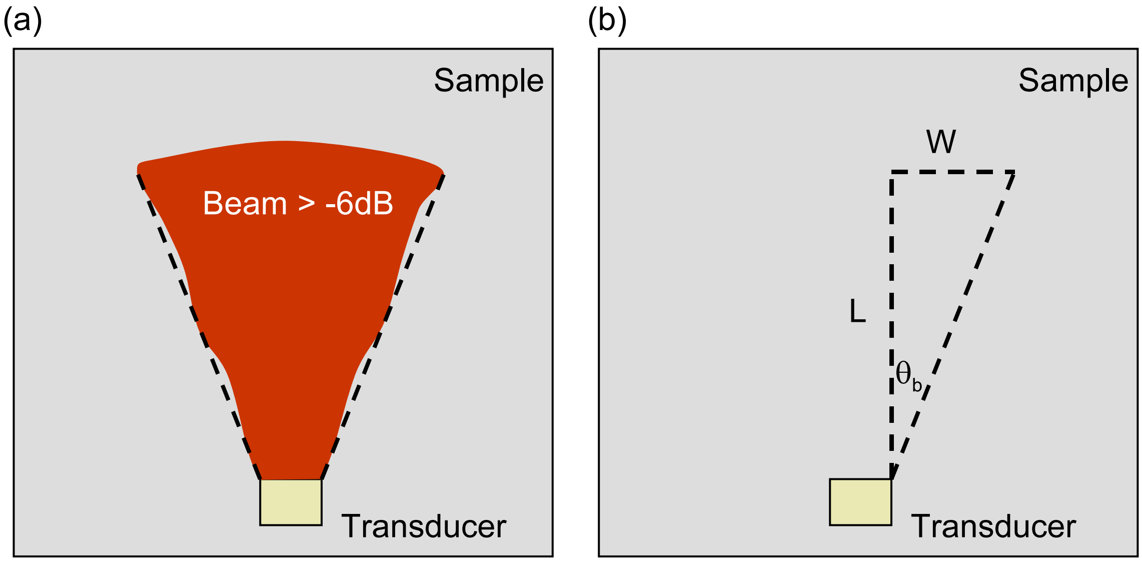


Figure 7.7: Schematic definition of beam divergence θ_b, showing a top-down view of a Lamb wave propagating forward from a transducer.

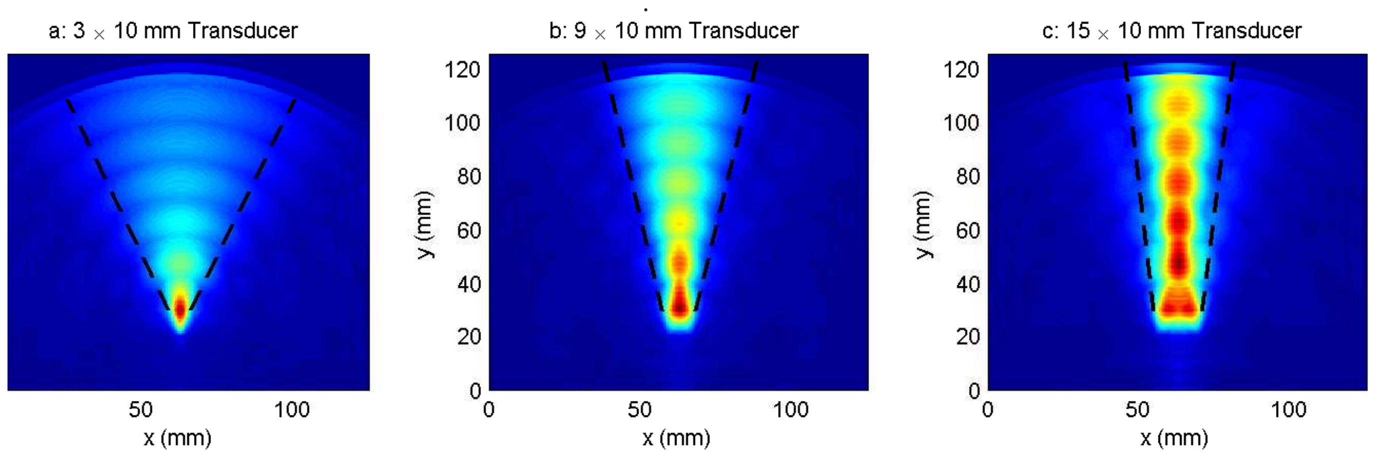


Figure 7.8: Simulated A_0 Lamb wave beam profile of 3, 9 and 15 mm wide air-coupled angled transducers. The beam divergence at -6dB (50%) is marked.

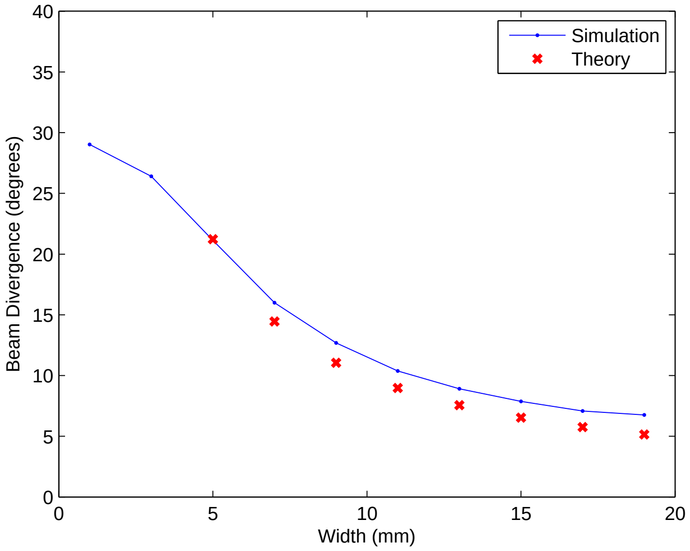


Figure 7.9: Beam divergence from Figure 7.5 plotted against transducer width. The theoretical results were calculated using Equation 7.9. The equation is only valid for transducer dimensions greater than the wavelength, so does not provided the divergence for transducers with a diameter of 1mm or 3mm.

Figure 7.10 shows a simulated Lamb wave B-Scan from a 3 × 10 mm transducer, using the same configuration as previously described. The resulting SAFT image is shown in Figure 7.11. The 3 × 10 mm transducer is better able to resolve the defect than the larger 15 × 10 mm transducer. A direct comparison is shown in Figure 7.21. Unfortunately, reducing the size of the transducer has a counter effect on system SNR.

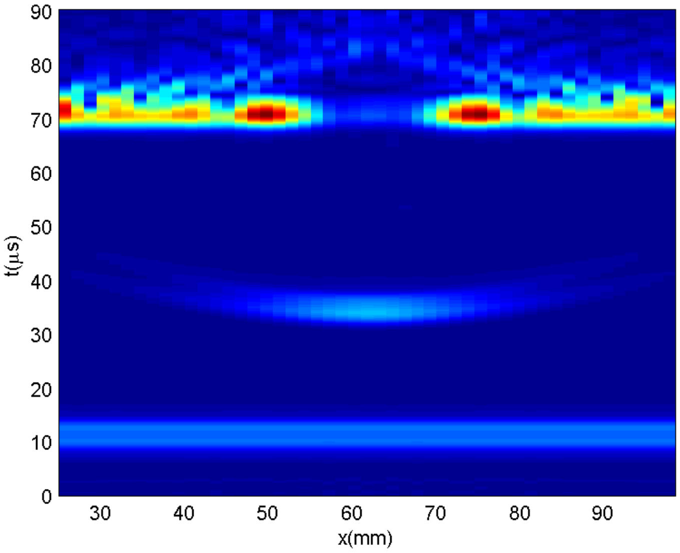


Figure 7.10: Repeated Figure 7.3 with smaller 3mm wide transducer

The amount of energy transferred into a sample is proportional to the transducer's surface area. Transducers with a small surface area generate and receive lower amplitude signals and hence suffer from lower SNR. In Chapter 4 30 × 30 mm transducers were used to achieve good SNR. Reducing a transducer's size from 30×30 mm to a 3×10 mm results in a 30dB reduction in energy input. In addition to this, the small transducer has a wider field of view, spreading the energy over a larger area. This leads to an additional reduction in Lamb wave magnitude. There will be a point when a reduction in SNR caused by smaller transducers cancels out the benefits of an increased field of view. Noise is considered in more detail in Section 7.4.

Farlow has produced air-coupled transducers as small as 7.5 mm^2 while still providing a useful SNR of 24dB. However, he used 20 cycle tone bursts that led to larger amplitude Lamb waves. This would provide poor resolution for pulse-echo imaging algorithms such as SAFT (20 cycle resolution = 70mm).

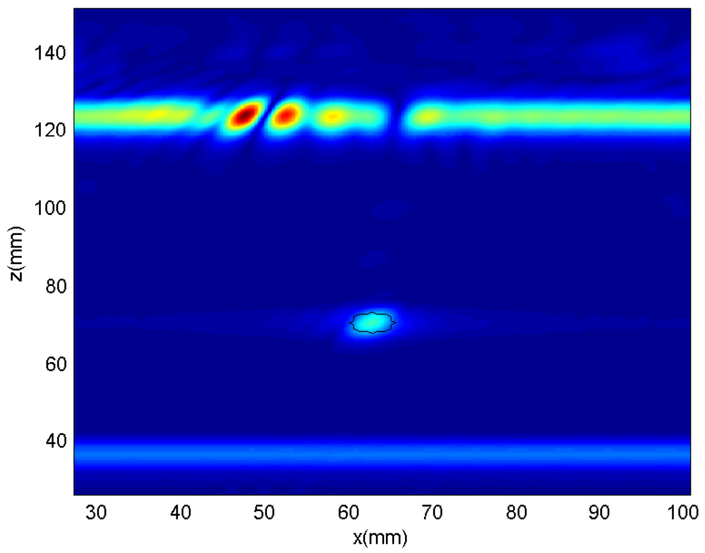


Figure 7.11: SAFT Processing of B-Scan in Figure 7.10. The -6dB defect outline is shown in Figure 7.21

One approach is to increase beam divergence, while maintaining surface area, is to reshape the transducer by increasing the length. Figure 7.12 shows the beam profile of three transducers, each 3mm wide, but with lengths of 10, 20 and 30 mm respectively. Figure 7.13 plots the beam divergence against length and Figure 7.14 shows a profile of the transducer beam at the point of maximum divergence. The beam divergence is not particularly sensitive to an increase in length. The beam divergence for 3 × 40 mm is 15.6^o rather than 9.63^o for a 12 × 10 mm transducer with the same surface area. Figure 7.21 shows the difference in the reconstruction of a defect by switching from a 15×10 mm transducer to a 5×30 mm transducer which both have the same surface area, hence energy input. The 5 × 30 mm transducer offers an improvement in the 'x' dimension, but a reduction in the 'y' dimension. This mirrors the change in transducer shape in each dimension.

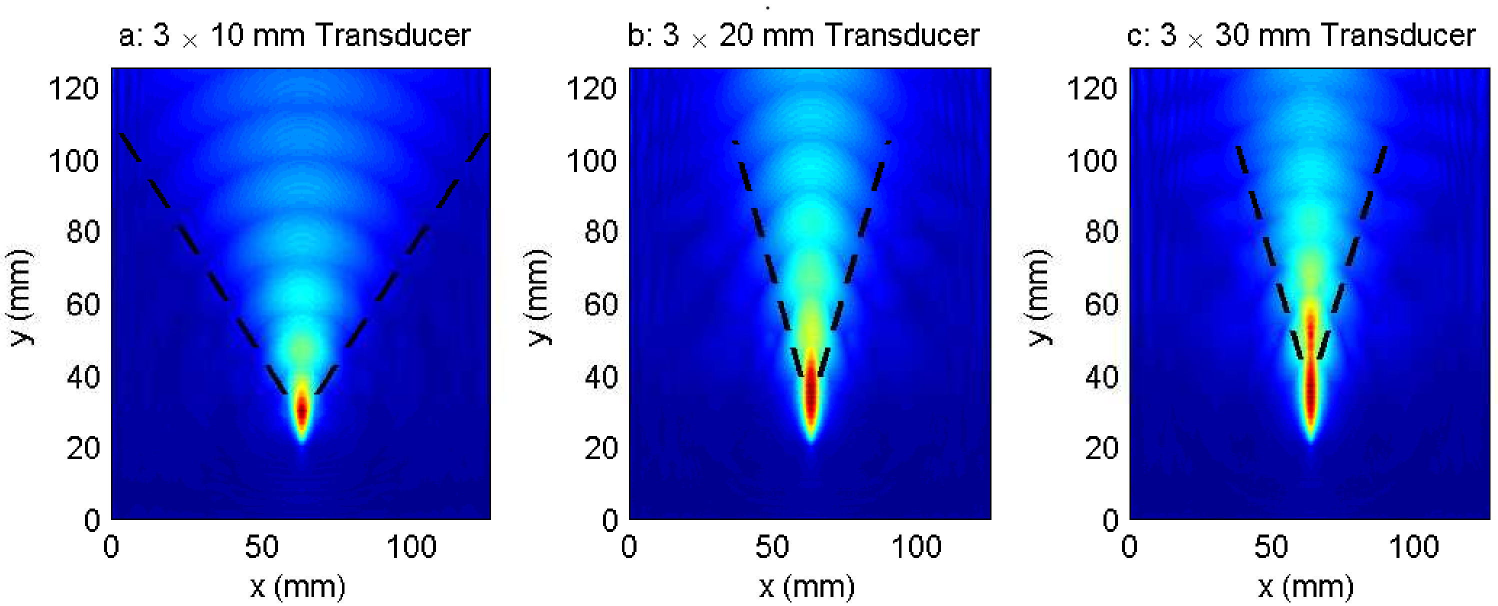


Figure 7.12: Simulated A_0 Lamb wave beam profile of 10, 20 and 30 mm long air-coupled angled transducers

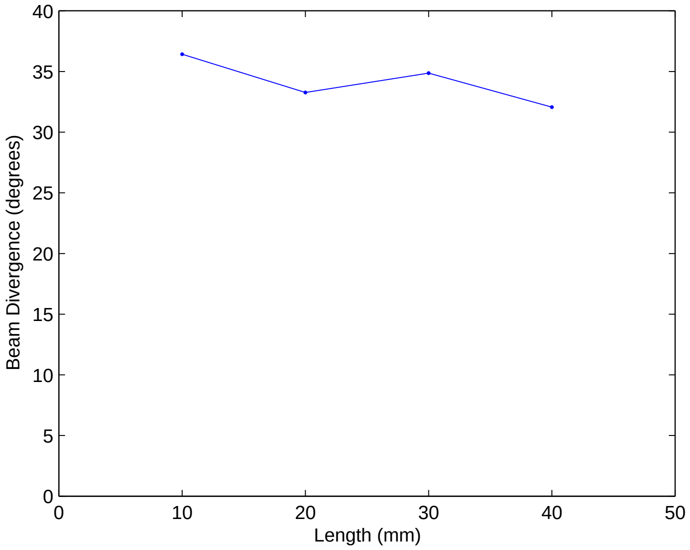


Figure 7.13: Beam divergence plotted against transducer length

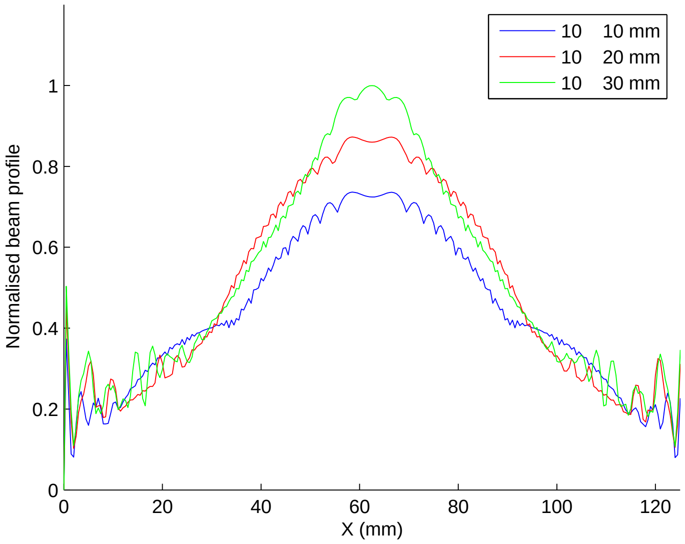


Figure 7.14: Beam profile of a horizontal cord positioned 62.5mm in front of the transducer, for a range of transducer lengths.

7.3 Modifying the Lamb Wave Beam Divergence

As discussed in the previous section, air-coupled angled Lamb wave transducers are very directional. For a 15 × 10 mm transducer, the beam divergence is in the order of 7.9^o (Figure 7.9). This limits the suitability of the transducers for any algorithm that expects a wide field of view, notably SAFT (Section 7.1), the Total Focusing Method [145] and Parallel Beam Tomography [142].

Beamforming [91] is a technique used to modify the beam characteristics of an acoustic, radio frequency (RF) or optical transducer. It is achieved by replacing the single element with an array of elements and applying appropriate phase delays to the individual elements to control the beam. It can be steered in a specific direction (Figure 7.15: b), focused to specific points (Figure 7.15: c) or defocused (Figure 7.15: d). Beamforming is used extensively in NDE and medical imaging. SAFT can be performed using a single array, transmitting and receiving on each individual element in turn. In this work, the imaging array is synthesised by physically positioning the transducer at each location. However, beamforming theory was applied to defocus the transducer element, giving it a wider field of view.

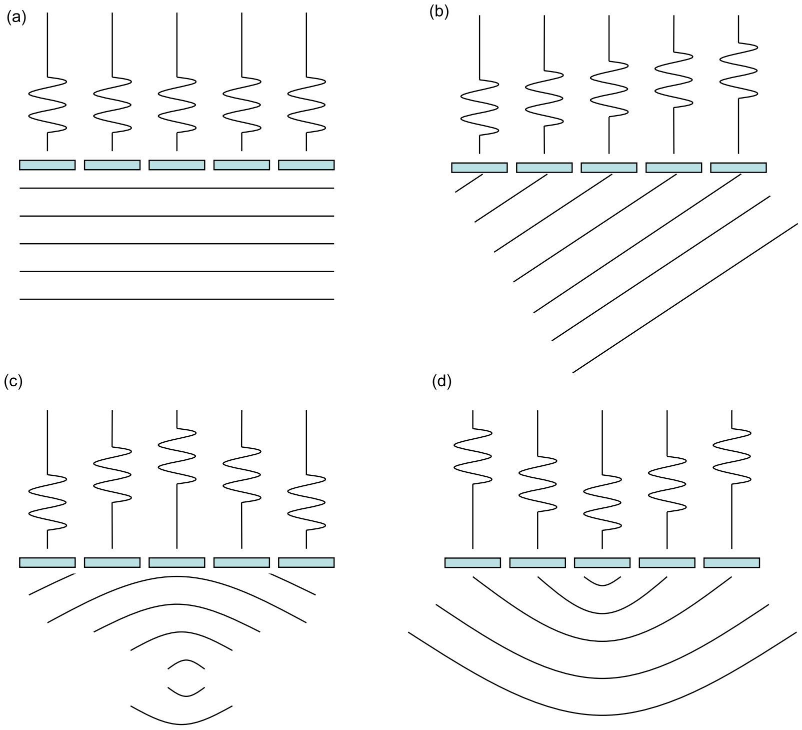


Figure 7.15: Schematic diagram of beamforming modalities, a: plane wave, b: beam steering, c: focusing and d: defocusing.

The air-coupled angled transducer was divided into a linear array of rectangular

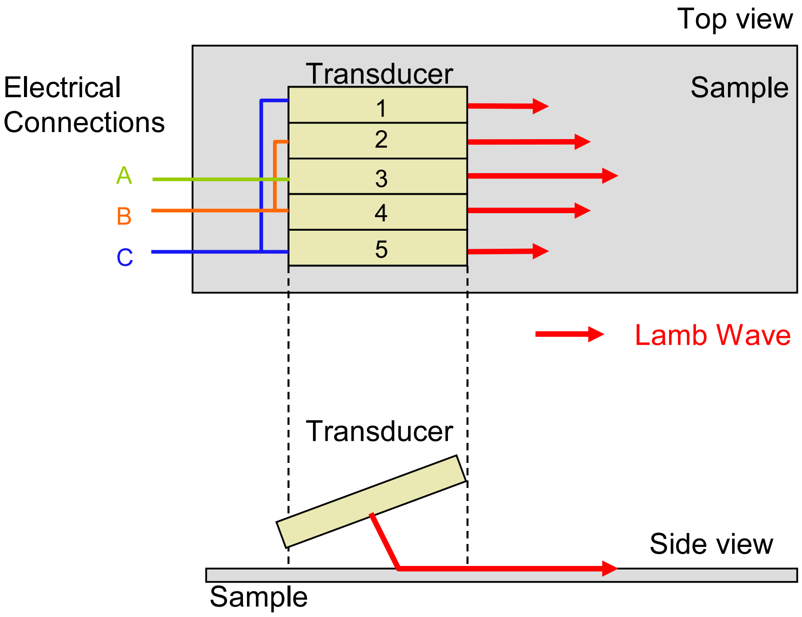


Figure 7.16: The air coupled transducer is divided into 5 array elements. Since the phase delays are symmetrical around the centre elements, adjacent elements can be electrically connected to simplify the electronics.

elements spaced at a maximum of half-wavelength spacing (based on the wavelength in the specimen). This is shown in Figure 7.16. The elements were excited with the appropriate delays to defocus the beam. The delays can be applied as either a time delay in the excitation signal, or spatially, by arranging the elements, moving the delayed elements back by the distance the wave would travel in the time delay period. Applying temporal delays offers a more flexible solution, but increases the driver electronics's complexity. Since the transducer is symmetrical, opposite elements can be connected, halving the number of parallel electronics channels. However, a limitation of this approach is that due to the dispersive nature of Lamb waves, the delays are only tuned for a specific phase velocity at the centre frequency. In addition the half wavelength spacing only applies for one frequency, but since air-coupled piezocomposite transducers are very narrow band, this is reasonable approximation. Figure 7.17 shows the geometry used to calculate the phase delays. The delays are shown in red, with d_1, d_5 being the longest delay and d_3 being equal to zero. The distances are converted to times (in seconds), using $t = \frac{d}{c}$ where d is the distance in meters and c is the phase velocity in meters per second. As θ was increased from 0^o to 15^o the beam divergence increased with an almost Gaussian shaped main lobe, as shown in Figure 7.18 (a) for $\theta = 15^o$. Beyond 15^o the beam pattern became non-uniform, breaking into three ridges, as evident in Figure 7.18 (b) for $\theta = 40^o$. Above 45^o the phase delays became too large and the beam became focused, as evident in

Figure 7.18 (c). Figure 7.19 shows the beam profile 62.5mm (y = 87.5mm) in front of the transducer. The widest divergence was at $\theta = 40^o$, as shown in Figure 7.20. Beamforming increased the beam divergence from 7.9^o degrees to 25.33^o for a 15 × 10 mm transducer.

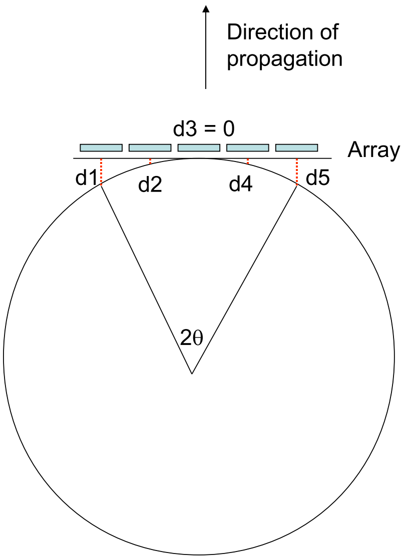


Figure 7.17: Definition of segment angle used to calculate phase delays.

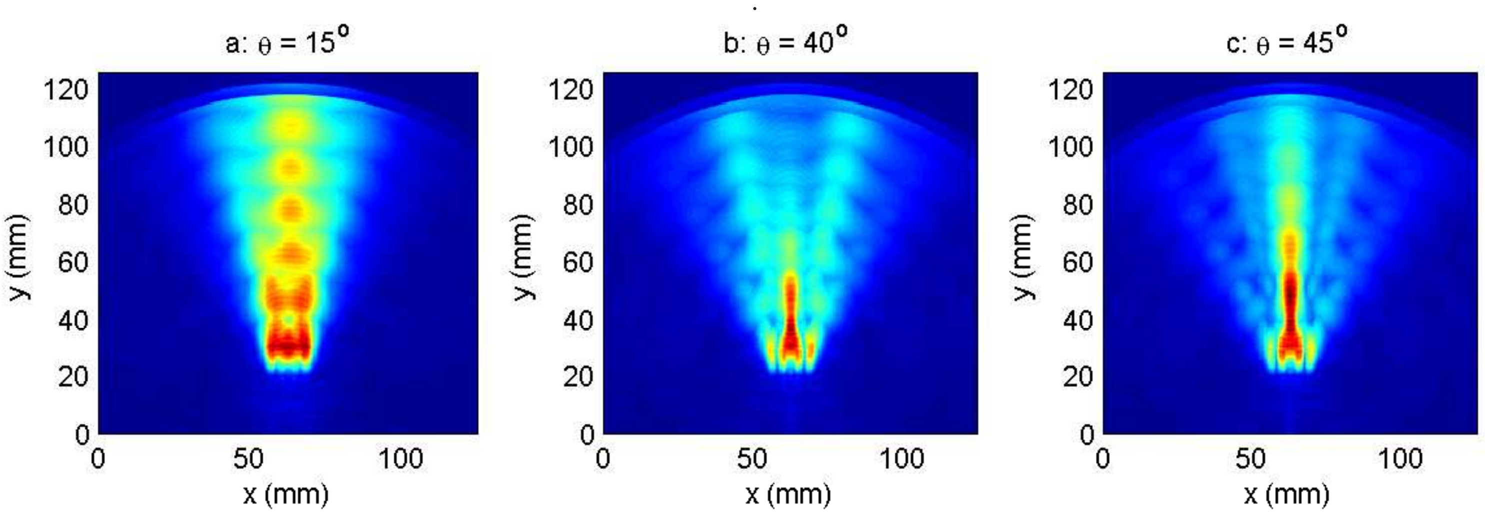


Figure 7.18: Simulated A_0 Lamb wave beam profile using 5 element linear array of angled transducers, with three different sets of delays derived from segment angles of 15^o, 40^o and 45^o as defined in Figure 7.17.

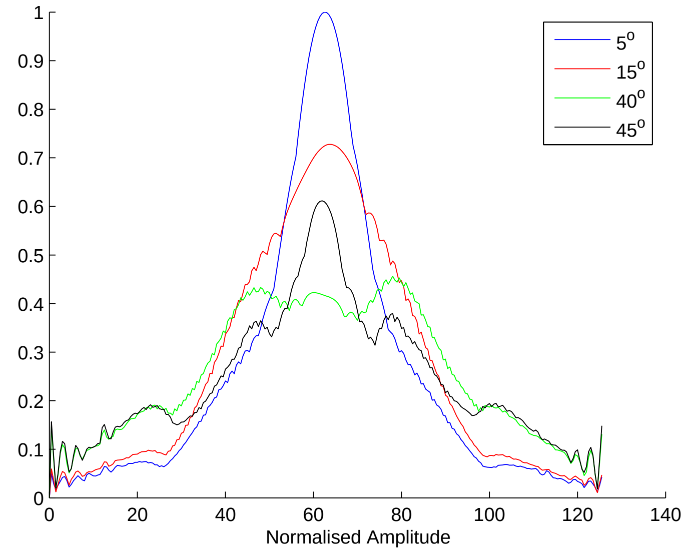


Figure 7.19: Beam profile of a horizontal cord positioned 62.5 mm in front of transducer (y = 87.5 mm) for several different segment angles (as defined in Figure 7.17).

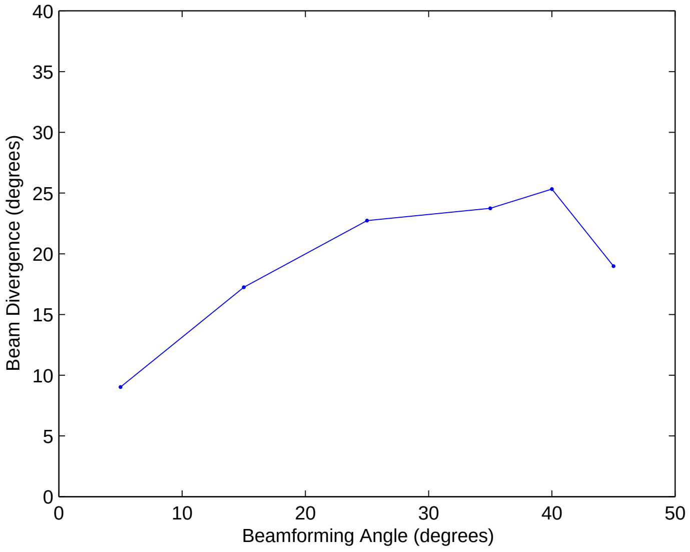


Figure 7.20: Beam divergence plotted against beamforming angle (as defined in Figure 7.17)

7.4 Comparison of Transducer Geometries

Figure 7.21 compares the SAFT reconstruction from four different transducers, the initial 15 × 10 mm, the smaller 3 × 10 mm, the reshaped 5 × 30 mm and the 15 × 10 mm five element array. Figure 7.22 shows the normalised receiver amplitude for each of the transducer configurations. Two measurements are taken from the array, the individual centre element and the electrically connected second and fourth elements. The small 3 × 10 mm transducer produces the most accurate defect reconstruction, but the receiver signal was of extremely low amplitude.

In practice, small receiver signals suffer from a reduction in SNR. This should be taken into account when comparing transducer designs. In this work, the noise was assumed to be constant for a single set of receiver electronics. The array transducer was realised by applying phased delayed excitations, as discussed in Section 7.3. In this case, the array required three sets of receiver electronics (See Figure 7.16), so the noise was modelled as three times greater than the single element case. Figure 7.23 shows the effect of adding Gaussian noise with a standard deviation of 0.2 to each trace - this was normalised against the maximum receiver amplitude of the 15 × 10 mm transducer, so had a standard deviation of 20% of the peak amplitude for the 15 × 10 mm transducer, but a standard deviation of 370% of the peak amplitude for the 3 × 10 mm transducer. In this case, the image deterioration due to noise, is proportional to the receiver signal size (Figure 7.22). Both the 15 × 10 mm transducer and the 5 × 30 mm transducer still produce reasonably good results, whereas the defect was not visible for the smaller 3 × 15 mm transducer. Of course, the noise value here was chosen arbitrarily and as the SNR of the small transducers is increased the results in Figure 7.23 will tend toward their no-noise counterparts (Figure 7.21) and the smaller transducer will thus give superior results.

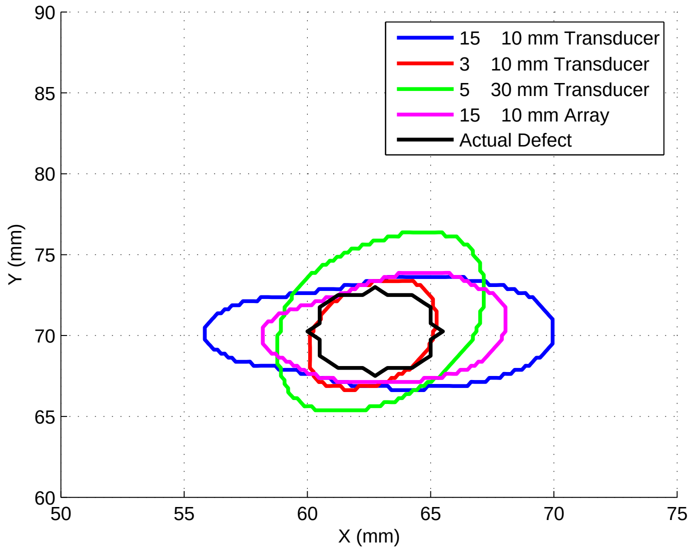


Figure 7.21: Profile of SAFT reconstruction from 4 different transducers.

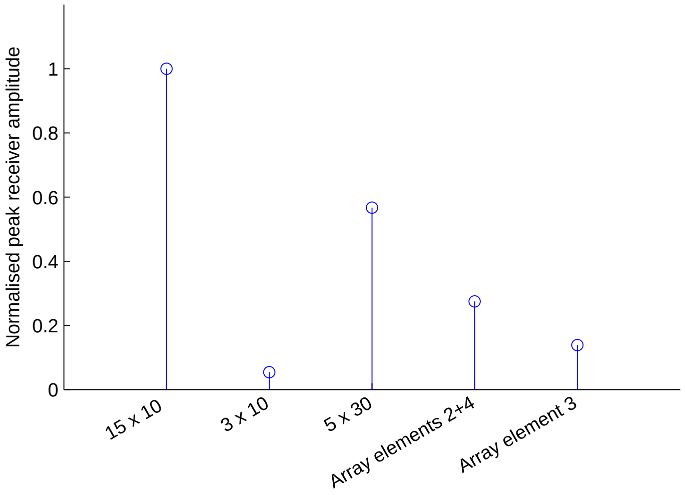


Figure 7.22: Sensitivity of five different transducer geometries.

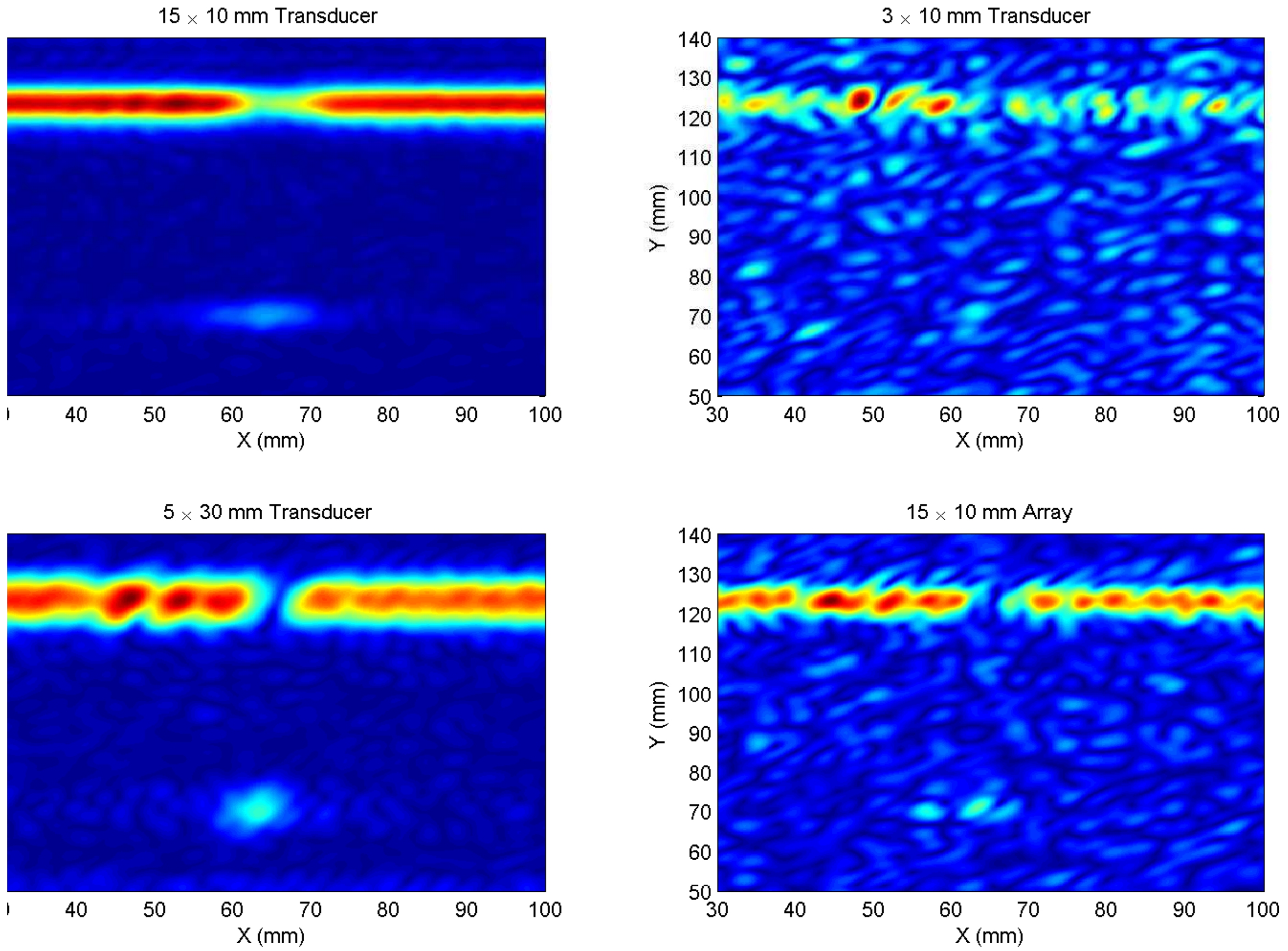


Figure 7.23: SAFT imaging using 4 transducer geometries, with constant noise

7.5 Positional Sensitivity

Small positional inaccuracies in transducer placement can become significant fractions of the wavelength and therefore distort phased compensation algorithms such as SAFT. The wavelength of the A_0 Lamb wave mode in 1mm aluminium at 600kHz is approximately 3.3mm ($\lambda = V/f = 2000/600k = 3\frac{1}{3}mm$). This is significantly smaller than the 10mm positional accuracy achieved in Chapter 3. This section investigates the effect of positional inaccuracy on SAFT reconstruction.

Consider a robot travelling in a straight line from the first element position to the last element position. The accuracy of each transducer position is not independent. In Chapter 3 the peak error in robot position was measured as approximately 15mm. However, two consecutive measurements intended to be 1.5mm apart were never 31.5mm apart (31.5mm inaccuracy would occur if two consecutive positions are at opposite extremes of ±15mm accuracy). Realistic measurement positions were calculated using the static model from Section 3.3.2, Chapter 3. These were used as transducer locations in the ultrasonic simulation. The positional error can be defined by the variance of the wheel rotation error, σ_w^2. The left and right wheel motion is calculated independently, so although statistically identical, there is a slight difference between the motion of each wheel. Figure 7.24 demonstrates examples of 75mm scans with varying severity of wheel speed variance from 0.001 mm^2 to 0.1 mm^2. In each scan, the transducer was positioned at each imprecise x,y, θ location. The results were processed using SAFT, using the same simulation configuration as previously described. The robot positioning simulation sample rate was 10ms. The simulated robot had a wheel diameter of 40mm and wheel base of 70mm. The mean wheel speed was set at 0.1875mm, which resulted in a speed of 7.5mm per second, or five 1.5mm spaced samples per second. At the most extreme variance of 0.1 mm^2 the standard deviation was 1.7 times the mean ($\sqrt{0.1} \simeq 1.7 \times 0.1875$).

Figure 7.25 shows SAFT reconstruction for wheel error variances of 0.001, 0.01 and 0.1 mm^2 respectively. The transducer had dimensions of 3 × 10 mm. The break down in image reconstruction is evident as the error in phase compensation prevents SAFT from perfectly reconstructing the image. In Figure 7.25, a threshold is applied at -6dB and the three scans superimposed as contour plots. There is a notable reduction in image quality above $0.01mm^2$. The experiments were repeated with the larger 10 × 15 mm transducer, with the results in Figure 7.27 and Figure 7.28, they show a similar trend and are equally sensitive to positional inaccuracies.

It is difficult to draw absolute conclusions from these results. Part of the deteri-

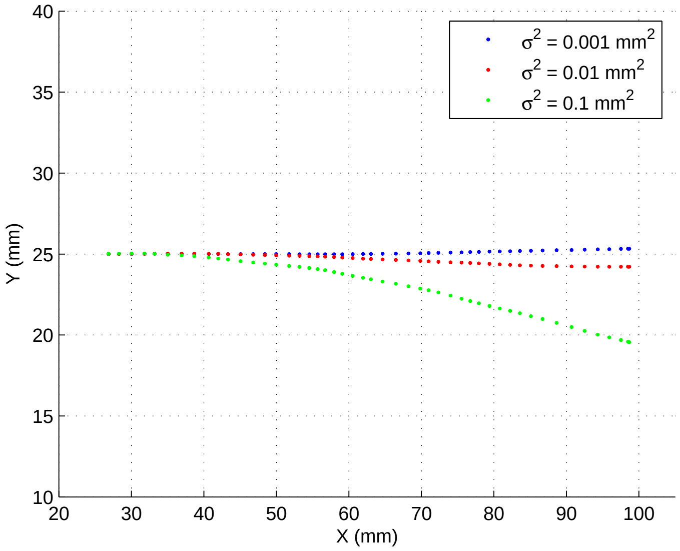


Figure 7.24: Deterioration of sensor positioning as robot wheel error variance increased from 0.001 mm^2 to 0.1 mm^2

oration in the images is due to inaccurate phase compensation, which is in turn related to transducer position inaccuracy, transducer field of view, the excitation wavelength and the defect's position in the sample. Defects near the transducers are only visible from a limited number of transducers. These will be less distorted than defects towards the extremities that are visible from a larger number of transducers, that require better positional accuracy to achieve phase compensation across all transducers. In addition to inaccurate phase compensation, there is a gradual shift in transducer misalignment as the robot deviates from the intended path, as evident in Figure 7.24 where $\sigma^2 = 0.1mm^2$ leading to a misplacement of defects. This is evident in the 'curved' back wall in Figure 7.29.

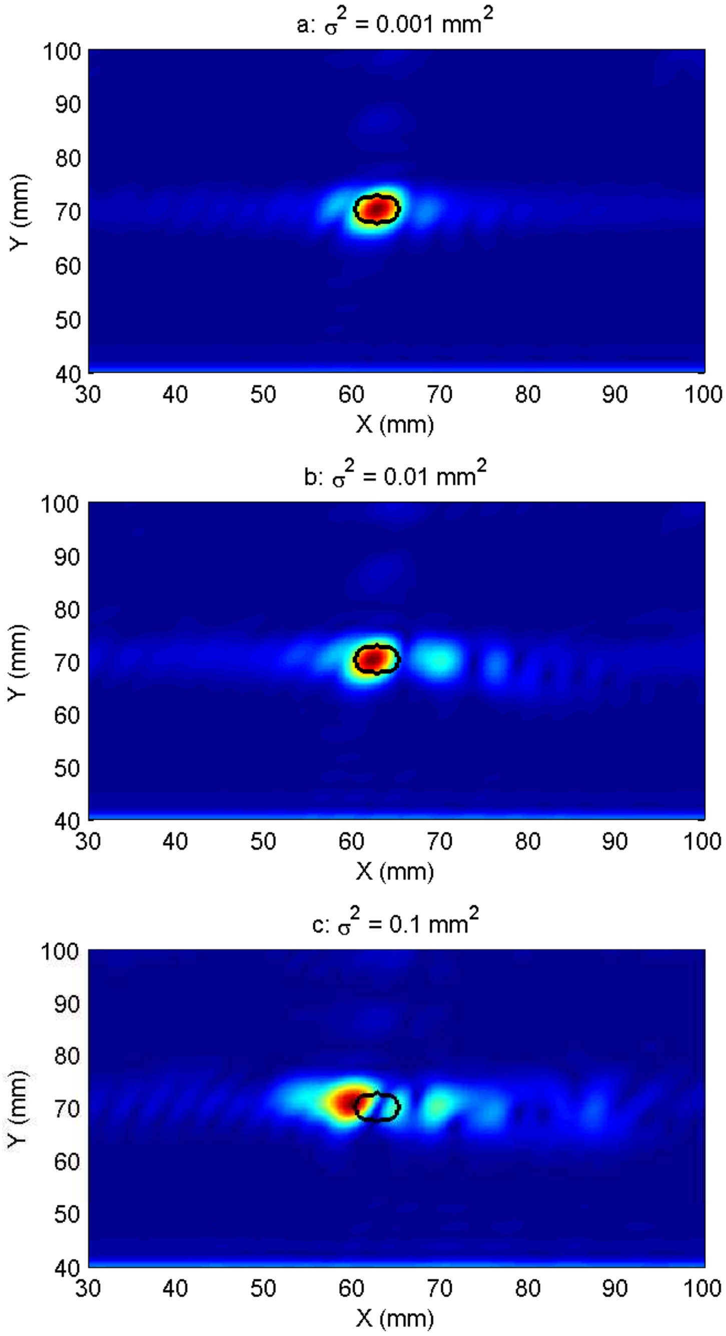


Figure 7.25: SAFT reconstruction for inaccurate transducer positions SAFT reconstruction for robot wheel error variances of 0.001, 0.01 and 0.1 mm^2. 3 × 10 mm transducer.

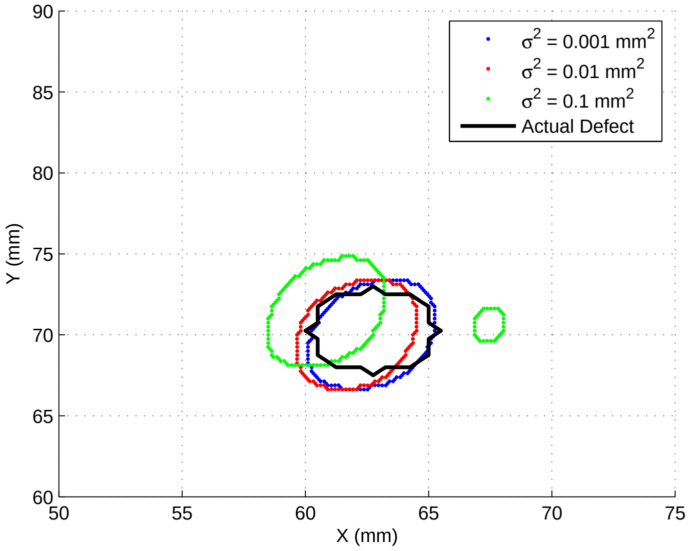


Figure 7.26: SAFT defect sizing for 3 × 10mm transducer positioned by a robot with wheel error variances ranging from 0.001 mm^2 to 0.1 mm^2.

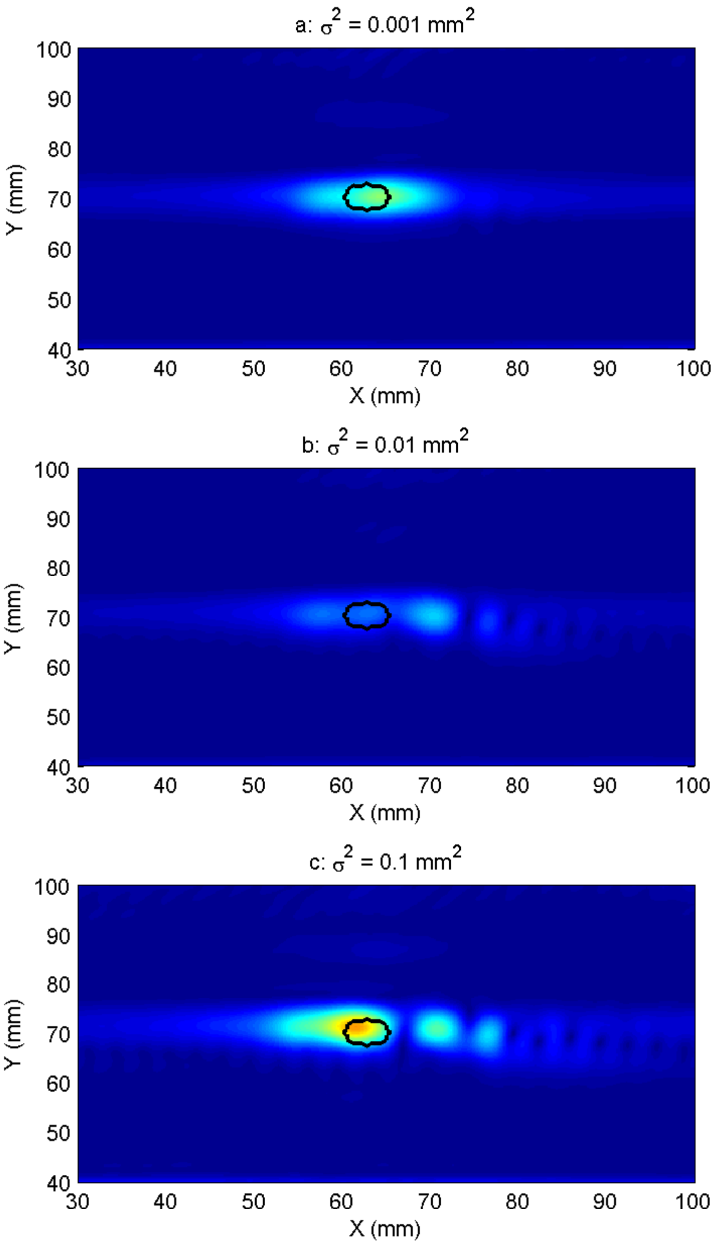


Figure 7.27: SAFT reconstruction for robot wheel error variances of 0.001, 0.01 and 0.1 mm^2. 15 × 10 mm transducer.

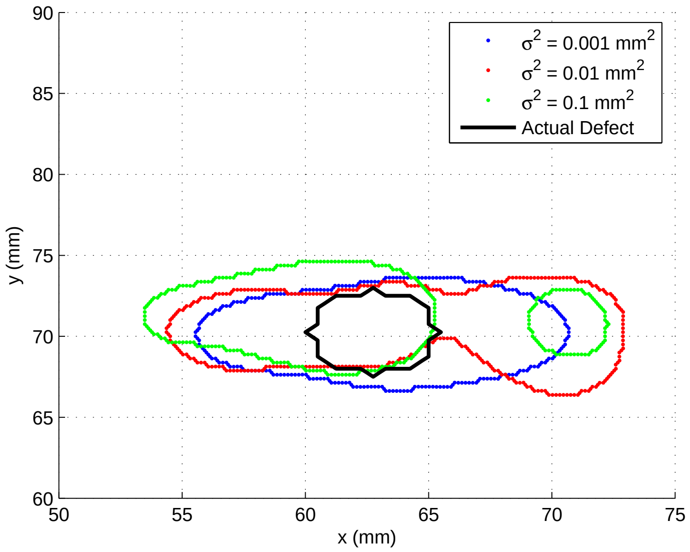


Figure 7.28: SAFT defect sizing for 15 × 10mm transducer positioned by a robot with wheel error variances ranging from 0.001 mm^2 to 0.1 mm^2.

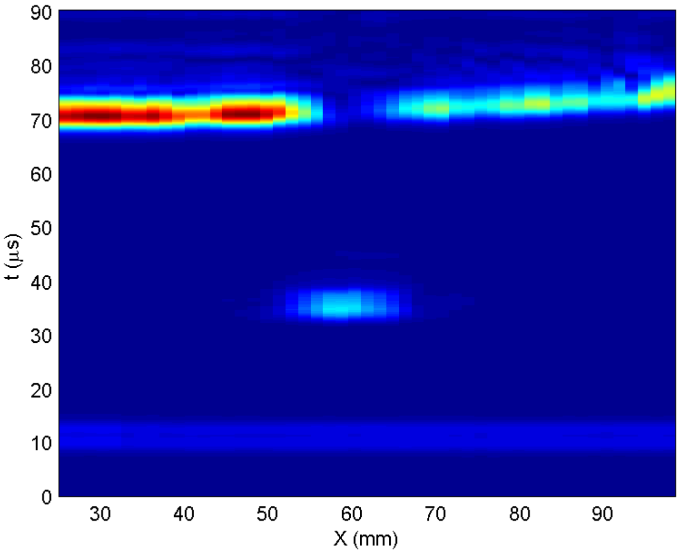


Figure 7.29: B-scan using a 15 × 10 mm transducer positioned by a robot with wheel error variance of 0.1 mm^2. The back wall appears 'curved' due to a gradual shift in transducer misalignment as shown in Figure 7.24.

By applying a condition in the summation of frequency components in Equation 7.7, it is possible to only include components that are within the transducer's field of view [148]. $\bar{S}(k_x, y = 0, \omega)$ from Equation 7.1 can be viewed as representing the acoustic field of frequency ω as a superposition of plane waves, incident at various angles, with components k_x and k_y. The field of view is defined as:

$$\sin(\theta) = \frac{k_x}{\frac{2\omega}{V_p}} \tag{7.10}$$

The field of view can be limited by only adding components that satisfy the inequality:

$$\omega > \frac{V_p}{2} \frac{k_x}{sin\theta_{view}} \tag{7.11}$$

Where θ_{view} is the field of view of the transducer measured from the centre outward. As 2θ approaches 180^o, this inequality reduces to include the frequency range $\frac{\Omega}{2} < \omega < \Omega$. Since the frequency domain data is symmetrical around $\frac{\Omega}{2}$ this essentially covers the entire range.

Limiting the field of view in the SAFT algorithm to match the transducer maximises SNR. This prevents the algorithm from including measurements where the pixel is outwith the view of the transducer, and have no other effect than to add noise.

Limiting the view in SAFT, beyond the transducer's physical field of view makes the algorithm less sensitive to errors in robot positioning. Consider Figure 7.31, when reconstructing the highlighted pixel, the narrow field of view in (a) only uses the information from three transducers. The transducers in (b) have a wider field of view, so information from five transducers is used. As the field of view increases, SAFT becomes more susceptible to a gradual change in positional error along the array. The trade-off is that using fewer measurements reduces the effectiveness of SAFT, thus reducing the resolving power. Figure 7.30 shows the results of limiting the field of view to 12^o for both the 15 × 10 mm transducer and the 3 × 10 transducer, with robot wheel variance of 0.1 mm^2. Comparing this to Figure 7.26 and Figure 7.28 shows an improvement in image quality, however both fall short of the ideal case in Figure 7.21.

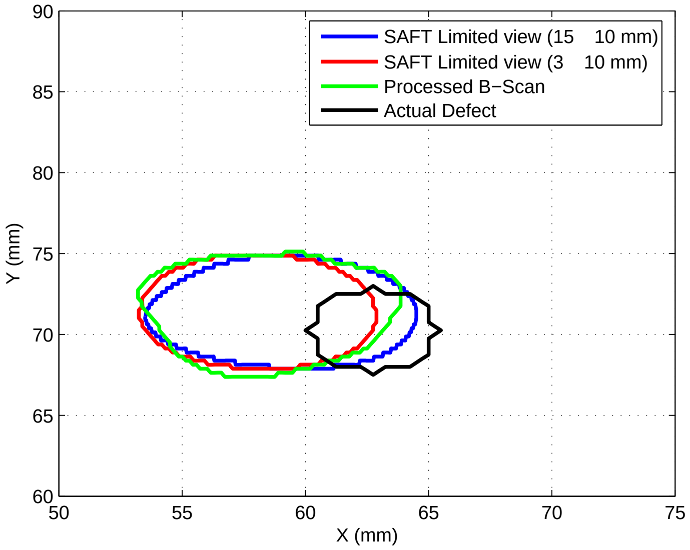


Figure 7.30: SAFT with limited field of view, compared with B-Scan algorithm. Robot had wheel inaccuracy of 0.1 mm^2

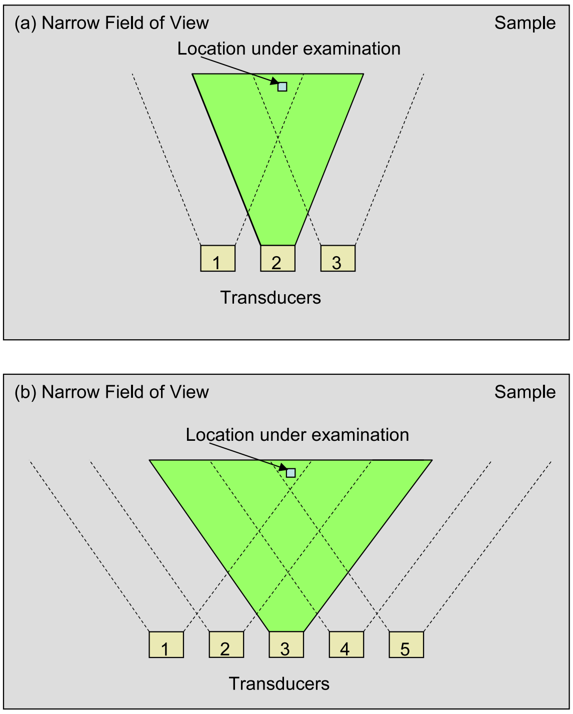


Figure 7.31: Schematic showing that as the field of view is reduced, less measurements are used in reconstruction

7.6 B-Scan Algorithm

If the transducer is very directional (such as the 15 × 10 mm transducer), then SAFT processing offers little benefit. The following algorithm processes the B-Scan, converting from $S(x, y = 0, t)$ to $B(x, y)$, but is assumed that each transducer is very directional. This is essentially the algorithm in Section 7.1 without the SAFT component. The algorithm is presented below:

1. For each transducer location, capture the pulse-echo time domain responses, $S(x = x_i, y = 0, t)$.

2. Convert the data to the frequency domain by taking the 1D Fourier transform of each time history.

$$\bar{S}(x = x_i, y = 0, \omega) = FFT_t\left[S(x = x_i, y = 0, t)\right] \tag{7.12}$$

3. Apply the back propagator (time delay)

$$\bar{S}(x = x_i, y, \omega) = \bar{S}(x = x_i, y = 0, \omega)\exp\left(iy\frac{2\omega}{V_p}\right) \tag{7.13}$$

An additional back propagator is applied for the angled wedge (in this case air). This takes a similar form to the previous back propagator:

$$P_W(y, \omega, \beta) = \exp\left(i\left[(1-\beta)y + \beta D_W\right]\frac{2\omega}{V_W}\right) \tag{7.14}$$

Where D_W is the distance the wave must travel in the wedge, and V_W is the phase velocity in the wedge (equal to the longitudinal velocity in the case of an air wedge). β is a binary operator that is 1 if $y < D_W$. It is used to toggle the expression for points inside and outside the wedge. The complete expression for $\bar{S}(x = x_i, y, \omega)$ becomes:

$$\bar{S}(x = x_i, y, \omega) = P_W(y, \omega, \beta)\bar{S}(x = x_i, y = 0, \omega)\exp\left(i\left\{y - D_W\right\}\beta\frac{2\omega}{V_p}\right) \tag{7.15}$$

D_W can be calculated as follows:

$$D_W = \frac{L_{tdr}}{2}tan(\theta_{tdr}) \tag{7.16}$$

where L_{tdr} is the length of the transducer and θ_{tdr} is the transducers incident angle (Section 5.6.2, Chapter 5).

4. Sum over the frequency components:

$$B(x = x_i, y) = \sum_{\omega \in \Omega} \bar{S}(x = x_i, y, \omega) \tag{7.17}$$

5. Finally, stack the slices $B(x = x_i, y)$, to create the 2D array B(x,y). Linear interpolation can be applied to improve the image resolution in the x dimension.

Figure 7.32 applies this algorithm to the inaccurately placed 15 × 10 mm transducer from Section 7.5 (robot wheel variance of 0.1 mm^2). The B-Scan and SAFT algorithms for a transducer with limited filed of view and inaccurate transducer positions are compared in Figure 7.30. The results are almost equivalent. In this case SAFT is not improving the reconstruction.

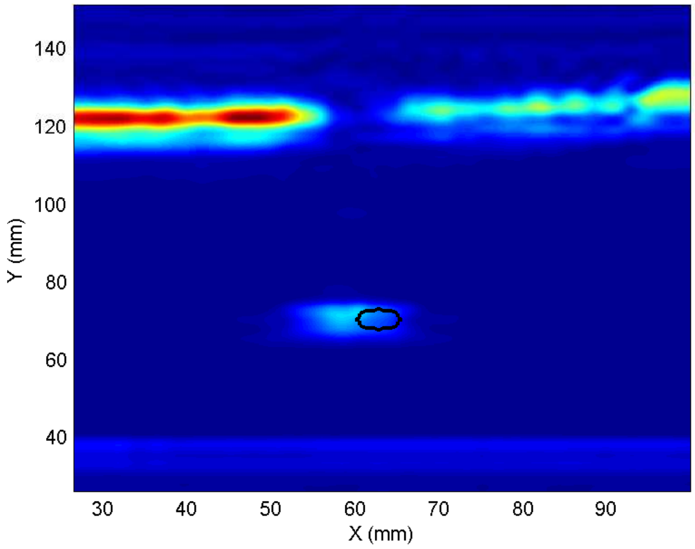


Figure 7.32: B-Scan reconstruction for robot wheel error variance of 0.1 mm^2. 15 × 10 mm transducer.

7.7 Operating Frequency

The last factor to be considered is transducer operating frequency. Lowering the operating frequency from 600kHz should increase the field of view and reduce the positional sensitivity of SAFT, but will lower resolution. Figure 7.33 shows the beam plot from a 15 × 10 mm transducer at an operating frequency of 200, 400 and 600kHz, using the optimal incident angle at each frequency. The beam divergence is plotted against frequency in Figure 7.34. Reducing the frequency from 600kHz to 200kHz increased the beam divergence from 8^o to 17^o, but also increased the wavelength from 3.3mm to 6.4mm (Figure 7.35). Good correlation is observed with theoretical results calculated using Equation 7.9.

Figure 7.36 shows the effect, on SAFT, of reducing the operating frequency from 600kHz to 200kHz. Figure 7.37 compares the -6dB defect sizing for the 15 × 10 mm transducer driven at both 600kHz and 200KHz. There is a slight improvement in resolving power in the X dimension, but a dramatic reduction in resolving power in the Y dimension. Although there will be an increased tolerance positional error, the effect is outweighed by the loss in resolution. To illustrate this, the defect sizing achieved with 600kHz transducers and relatively extreme robot wheel error of $0.1mm^2$, shown in Figure 7.30, exceeds the results achieved with the perfectly placed 200KHz transducers shown in Figure 7.36.

It should be noted that both SAFT and the B-Scan algorithm counter the effects of dispersion, by considering each frequency component in turn. Therefore changing the operating frequency and hence, region on the dispersion curve, should not influence results.

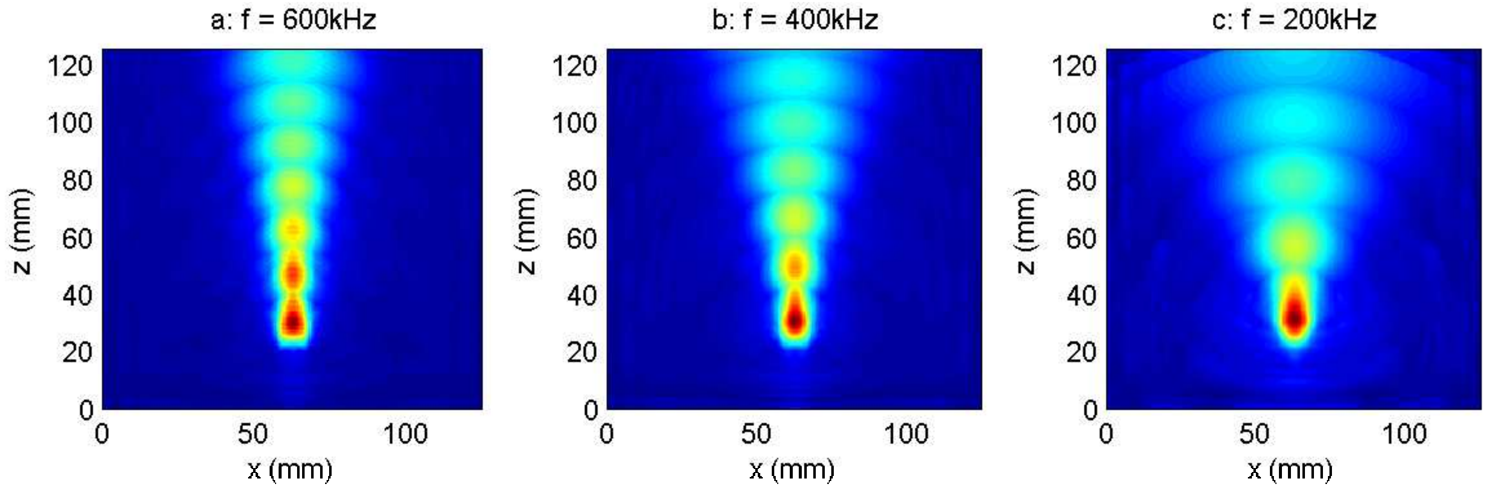


Figure 7.33: Beam profile at different excitation frequencies, 10 × 10 mm transducer

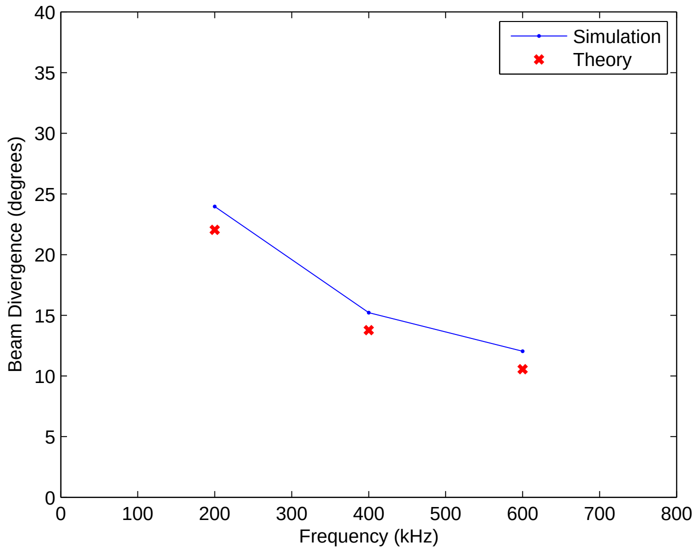


Figure 7.34: Beam divergence plotted against frequency

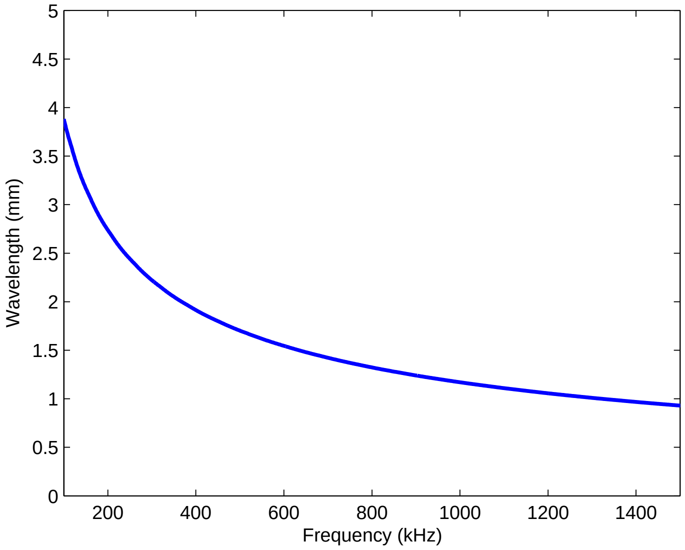


Figure 7.35: Wavelength plotted against frequency for A_0

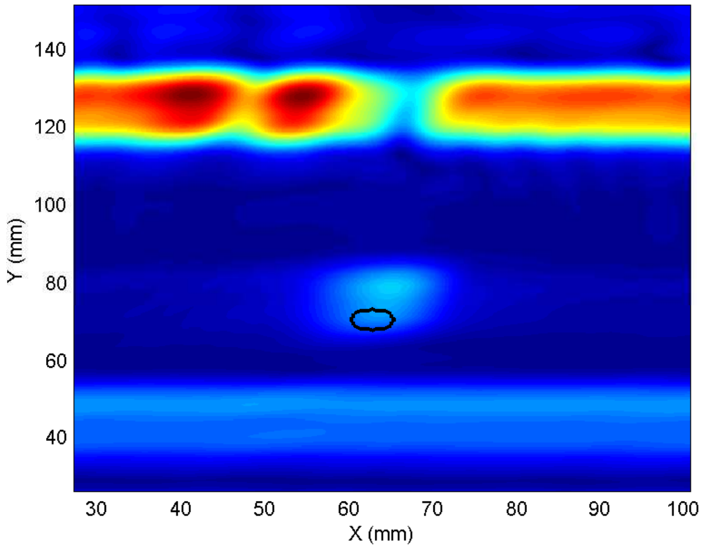


Figure 7.36: SAFT processed image, 200kHz 10 × 10 mm transducer

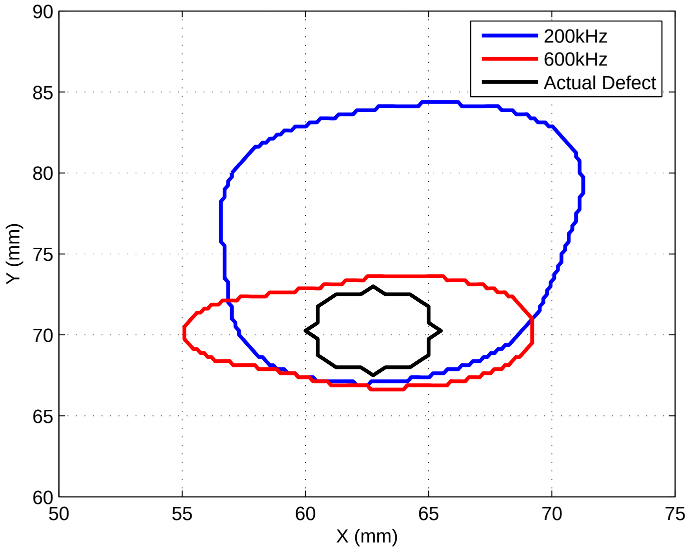


Figure 7.37: Effect of reducing frequency from 600kHz to 200kHz

7.8 SAFT Summary

The applicability of SAFT for imaging using miniature robotic vehicles has been considered. The two key factors have been identified; transducer beam divergence and transducer positioning.

As the beam divergence increases, reflectors become visible to more transducers, increasing the effectiveness of SAFT in terms of resolving power and SNR. Beam divergence can be increased by reducing the width of the transducer. Small transducers suffer from poor SNR which in turn reduces the quality of the image. Transducers can be defocused using beamforming. This gives them a wider field of view, improving their suitability to SAFT processing. This can be achieved electrically, by time delaying channels or mechanically, by physically arranging the transducer elements. The electrical technique is more adaptable, but results in low amplitude receiver signals, and hence low SNR. Lowering the operating frequency increases beam divergence, but increases the wavelength, which in turn reduces image fidelity.

SAFT's phase compensation is sensitive to transducer positional inaccuracies which introduce unpredictable phase shifts. Reducing the field of view in SAFT, reduces the number of measurements involved in reconstruction at any point, reducing the positional sensitivity. However, this limits the effectiveness of the SAFT algorithm in terms of resolving power and SNR. In addition to this, defect positioning is relative to transducer positions, so poor placement of transducers leads to a poor estimation of defect position.
The following guidelines summarise the applicability of SAFT:

- If the transducer has a wide beam divergence ($> 15^o$) and the positional accuracy is good (${\sigma_w}^2 < 0.01mm^2$), best results are achieved using SAFT.
- If the transducer has a wide beam divergence ($> 15^o$) and the positional accuracy is poor (${\sigma_w}^2 > 0.05mm^2$), best results are achieved using SAFT with limited field of view.
- If the transducer has a narrow beam divergence ($< 10^o$) and the positional accuracy is good (${\sigma_w}^2 < 0.01mm^2$), SAFT offers no advantage.
- If the transducer has a narrow beam divergence ($< 10^o$) and the positional accuracy is poor (${\sigma_w}^2 > 0.05mm^2$), SAFT with a limited field of view offers no advantage, SAFT without a limited view causes a deterioration in the image.

7.8.1 Fundamental Limitations of SAFT

The dominant assumption in SAFT is that a defect can be represented by a number of single non-interacting scalar point scatters. Consider a plate with one defect at y = 100mm and another at y = 150mm. The reflection from the defect at y = 150mm, will also be influenced by the defect at y = 100mm. This leads to blurring of the resulting image. In practice, one part of a single defect, such as a crack, will influence the imaging of another part, leading to a loss in fidelity.

Defects approaching 100% depth will cause a shadow that cannot be penetrated by SAFT, potentially masking defects.

Extended discontinuities such as plate edges or artificial defects do not always reflect energy back to the transducer, so are not satisfactorily detected. This is the reason that the left and right plate edges are missing from all B-Scan and SAFT reconstructions, such as Figure 7.10.

In addition to this, Lamb wave phase velocity is dependent on plate thickness, so must be estimated for each sample. This approximation is only practical for uniform thickness geometries with a smaller number of defects. Local variations in Lamb wave velocity due to a change in thickness causes a deterioration in SAFT imaging.

7.9 Experimental B-Scan

A 300 × 300 × 1 mm aluminium plate section was scanned using the robot described in Chapter 2, carrying the payload described in Chapter 4. The plate had a 20 × 20 mm artificial defect, near the centre. The defect was 0.5mm deep. The robot was instructed to scan the sample with a resolution of 1mm. The transducers were 30 × 30 mm, so had minimal beam divergence. the B-scan algorithm discussed in Section 7.6 was applied to reconstruct the data.

Figure 7.38 shows the resulting image. The defect and back wall are clearly visible. The defect clearly casts a shadow on the back wall. Figure 7.39 shows the region around the defect in more detail. Positional inaccuracy has had minimal effect on image quality, with the defect appearing in the correct location (Figure 7.39) and the back wall appearing straight (Figure 7.38).

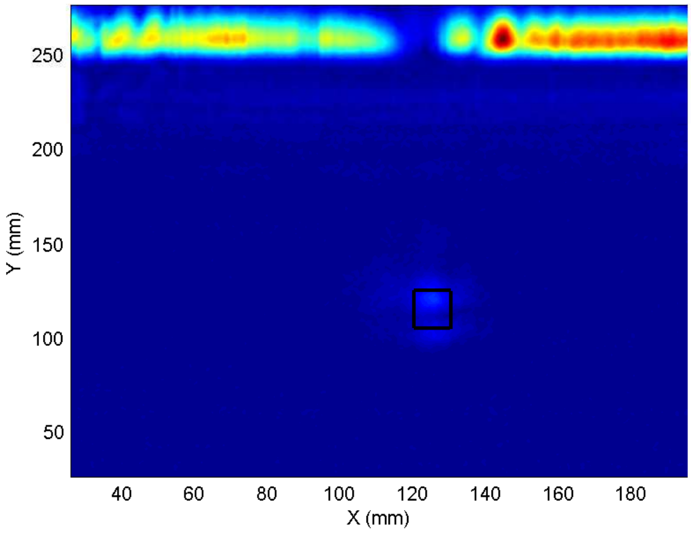


Figure 7.38: Experimental B-scan of 1mm thick aluminium plate with a 20 × 20 mm artificial defect using the ultrasonic payload discussed in Chapter 4. The defect is shown in black. It is 50% of the plate thickness

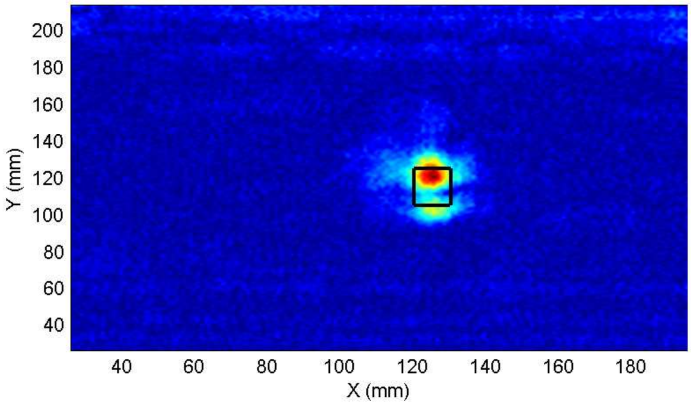


Figure 7.39: Magnified section of Figure 7.38

7.10 Alternative Imaging Techniques

7.10.1 Total Focusing Method

Phased array imaging is a logical extension to SAFT. Instead of performing simultaneous pulse-echo calculations with each element in turn, the phased array simultaneously transmits and receives on all elements. In a process known as the **T**otal **F**ocusing **M**ethod, **(**TFM) [145], beamforming is used to focus the ultrasonic beam on each pixel in the resulting image. Karaman et al. [146] have evaluated different synthetic aperture and phased array techniques for medical imaging and found that for a 128 element array, SAFT typically had a contrast to noise ratio of 30% of the equivalent phased array using TFM. However, they found that they could achieve 80% of the contrast-to-noise ratio of a phased array by performing multi-element SAFT. They transmitted on a series of 12 elements, defocusing the beam in a manner similar to Section 7.3. This 12 element sub-array was moved along the full 128 element array. Reception was performed in parallel over the same 12 element sub-array.

It is not practical to directly implement a phased array imaging strategy in the miniature robotic vehicle platform. It would require a large number of robots positioned very close together ($\lambda/2$ spacing $\simeq$ 1.6mm). However, the approach can be synthesised using **F**ull **M**atrix **C**apture [145] (**FMC**). FMC captures all transmit-receive pairs sequentially (N^2 measurements where N is the array length). Phased array imaging can then be performed in post processing. Since the transmit elements fire sequentially, there is $1/N$ less energy put into the system for each measurement than with a phased array. This has the side effect of reducing SNR.

In practice FMC could be performed by two or more robots. One robot would be responsible for ultrasonic transmission, it would move in a single straight line path along one edge of the sample. One or more additional robots would be responsible for ultrasonic measurement. These robots would measure the propagated wave at each of the receiver locations, for each transmission location. The total focusing method has the same fundamental limitations of SAFT detailed in Section 7.8.1.

7.10.2 Projection Tomography

Projection tomography can recover the image of an object's cross section from a series of projections. The projections are considered integrations over straight lines through the object. The key to image reconstruction is the Fourier Slice Theorem [141] which relates the projection data to the 2D Fourier transform of the object's cross section. The principle of Fourier Slice Theorem is shown in Figure 7.40 a. It is possible to generate a reasonable approximation of the 2D Fourier domain from a series of projections, as shown in 7.40 b. The frequency domain information is then converted back into the space domain to reveal the image. In order to use an inverse Fourier transform, the data must be interpolated to a regular grid. It is difficult to interpolate adequately in the frequency domain, so an approach called the Filter Back Propagation Algorithm [141] is generally used. The Filter Back Propagation Algorithm offers two advantages over frequency domain interpolation. Firstly, interpolation can be done in the space domain, which is usually more accurate. Secondly, the projections can be processed sequentially as they arrive, increasing speed and reducing memory requirements. The algorithm is shown below; a derivation can be found in [141].

- Measure the projection $P_\theta(n\tau)$ where τ is the spatial sample interval as shown in Figure 7.40.
- Design a frequency domain filter $H(\omega)$ as shown in Figure 7.40 c. This is defined as $H(\omega) = FFT(h(n\tau))$ where:

$$h(n\tau) = \begin{cases} 1/4\tau^2, & n = 0 \\ 0, & \text{n even} \\ \frac{-1}{n^2\tau^2\pi^2}, & \text{n odd} \end{cases} \tag{7.18}$$

- The filtered propagation at the sample points is calculated as:

$$Q_\theta(n\tau) = IFFT(FFT(P_\theta(n\tau)) \times FFT(h(n\tau))) \tag{7.19}$$

- Take the inverse Fourier transform of each projection and perform the back propagation:

$$f(x,y) = \frac{\pi}{K} \sum_{i=1}^{K} Q_{\theta i}(x\cos\theta_i + y\sin\theta_i) \tag{7.20}$$

  Q_θ does not typically fall on a sample point $Q_\theta(n\tau)$, so is calculated from linear interpolation.

The projection may be either signal attenuation, or transit time, both of which will be altered by a defect. Figure 7.41 shows an example of straight-ray tomography

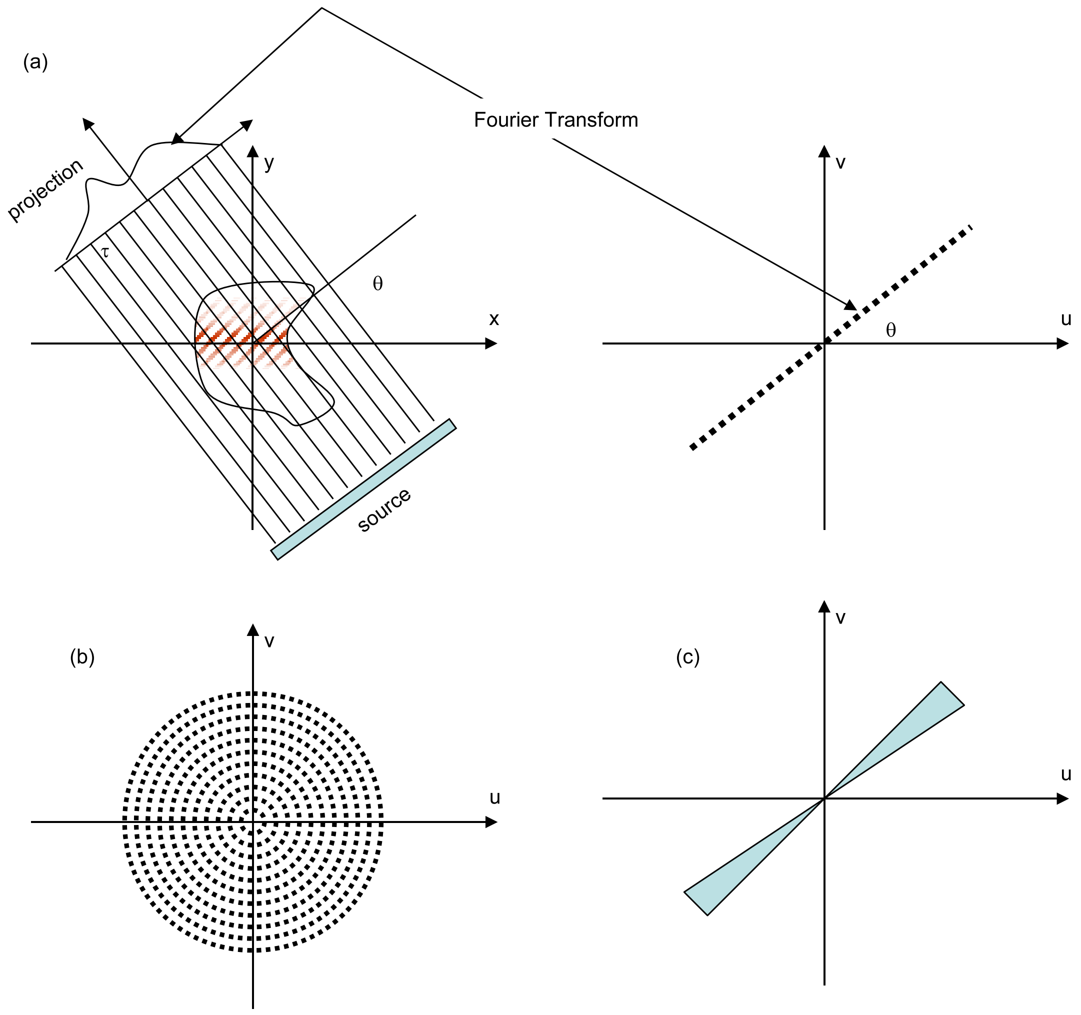


Figure 7.40: (a) shows the Fourier Slice Theorem. (b) shows how a series of samples can give complete coverage of the frequency domain. (c) shows the filter $H(\omega)$ used to weight the projection relative to it's density in the frequency domain

performed on a 125 × 125 mm, 1mm aluminium sample. The projections were generated using the simulation tool described in Chapter 5. There were 72 projections at 5^o increments. The sample spacing was 1.5mm (59 pitch-catch measurements); the entire scan required 4248 measurements. The air-coupled transducers were held at the incident angle required to generate the A_0 Lamb wave mode. The transducers had dimensions of 30 × 30 mm and were excited with a 10 cycle, 600kHz tone burst.

The projection of signal amplitude gave better results than time of flight. This was due to the difficulty in accurately measuring the time of flight of a dispersive signal. Both defects are clearly visible, yet both are oversized. For comparison, Figure 7.42 shows the same measurements reconstructed using the time of flight information.

The dominant approximation in the Fourier Slice Theorem is that the energy travels in straight lines through the sample. This is a poor approximation for diffracting sources, such as ultrasound. The straight ray approximation is analysed in detail in [143], but in general it tends to make defects appear larger, often with a null in the centre. This approximation can be overcome by extending the theory to Diffraction Tomography.

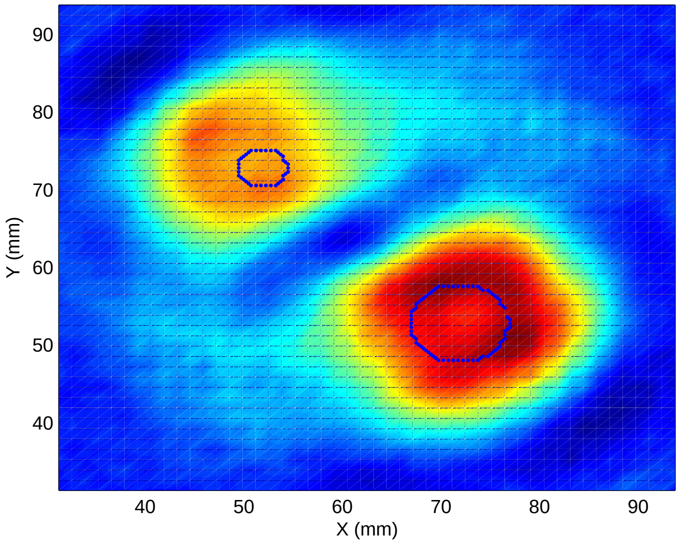


Figure 7.41: Straight Ray Projection Tomography of 62.5 × 62.5 mm region in 125 × 125 mm plate with two 50% thinned regions. Signal attenuation projection.

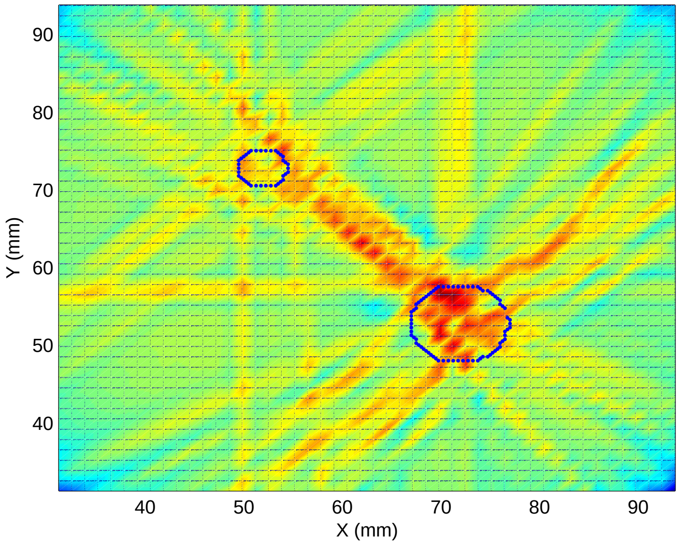


Figure 7.42: Straight Ray Projection Tomography of 62.5 × 62.5 mm region in 125 × 125 mm plate with two 50% thinned regions. Time of flight projection.

7.10.3 Diffraction Tomography

The Fourier Diffraction Theorem [141] relates forward scattering data with the Fourier transform of a cross section of the object. It is similar to the Fourier Projection Theorem, but instead of the projection forming a straight line in Fourier space, it forms an arc. This is shown in Figure 7.43. The radius of the arc is equal to k_0 defined as:

$$k_o = \frac{2\pi}{\lambda} \tag{7.21}$$

where λ = wavelength.

Multiple frequencies can be used to generate arcs of different radii, giving better coverage of the frequency domain. As the wavelength of the excitation signal tends towards zero, the radius tends to infinity and the Fourier Diffraction Theorem tends towards the Fourier Projection Theorem.

Diffraction tomography relies on an approximation to the governing wave equation, typically the first Born or first Rytov approximation [141]. Both of these approximations assume weak scattering. The Born approximation breaks down with features that are significantly larger than a wavelength. The Rytov approximation breaks down when the refractive index varies by more than a few percent.

The computational expense of space domain interpolation of diffraction projections makes frequency domain interpolation the preferred approach for diffraction tomography [141]. Recently, the non-uniform FFT has shown some promise as an alternative to interpolation [150]. The implementation of a diffraction tomography algorithm fell out of the scope of this work.

7.10.4 Double Crosshole Lamb Wave Tomography

In algebraic tomography, the image is considered to be an array of unknowns that can be solved algebraically. Unlike projection tomography, it does not require as large a number of projections or evenly spaced projections over 180^o or 360^o. Although conceptually much simpler than projection tomography, algebraic tomography lacks the fidelity and speed of implementation [141].

Malyarenko and Hinders [142] have demonstrated Double Crosshole Lamb wave tomography using the S_0 Lamb wave mode. Figure 7.44 shows a schematic of the experiment. The sample is divided into an N × N grid of cells. The experiment records the time of flight from each of the transducers along the bottom (i = 1 to N) to each of the receivers along the top (j = 1 to N), capturing N^2 pitch-catch measurements. The sample is rotated by 90^o and the process is repeated to capture an additional N^2 measurements. The transducers are typically radial sources. Reflections from plate edges are not a problem since they always arrive later than the wave travelling directly from transmitter to receiver. The data is generally reconstructed using the **S**imultaneous **I**terative **R**econstruction **T**echniques (**SIRT**) [141]. SIRT [141] iteratively solves for the contribution of each cell using the measurements across the grid in many different directions. In this case, SIRT provides a transit time for each cell. The material thickness can be inferred from the transit time, since Lamb wave velocity is frequency-thickness dependent and the frequency is constant. In addition to this, the wave may travel around defects, resulting in a delay. In theory, signal attenuation can be used as well as time of flight measurements, but this is complicated by propagation paths of varying length, which results in uneven signal attenuation.

The double cross hole approach is appealing for use on miniature robotic vehicles for several reasons.

1. It can deal with projections from arbitrary positions.

2. All measurements are captured separately, so there is scope to run the measurement process in parallel using multiple robots.

3. The iterative nature of the reconstruction algorithm could allow the robots to focus on specific areas as the experiment progressed, skipping measurements across sections of the sample that were showing cause for concern.

However, there are several issues.

1. The double cross hole approach requires $2 \times N^2$ measurements. For a 75mm × 75 mm sample, with 1.5mm spacing 1250 measurements are required, in comparison to 50 for SAFT.

2. The transducers must have wide beam divergence so that an excitation at transducer i=1 is detected by a receiver at j=N. This is not ideal for air-coupled angular transducers which are typically very directional (see Section 7.2).

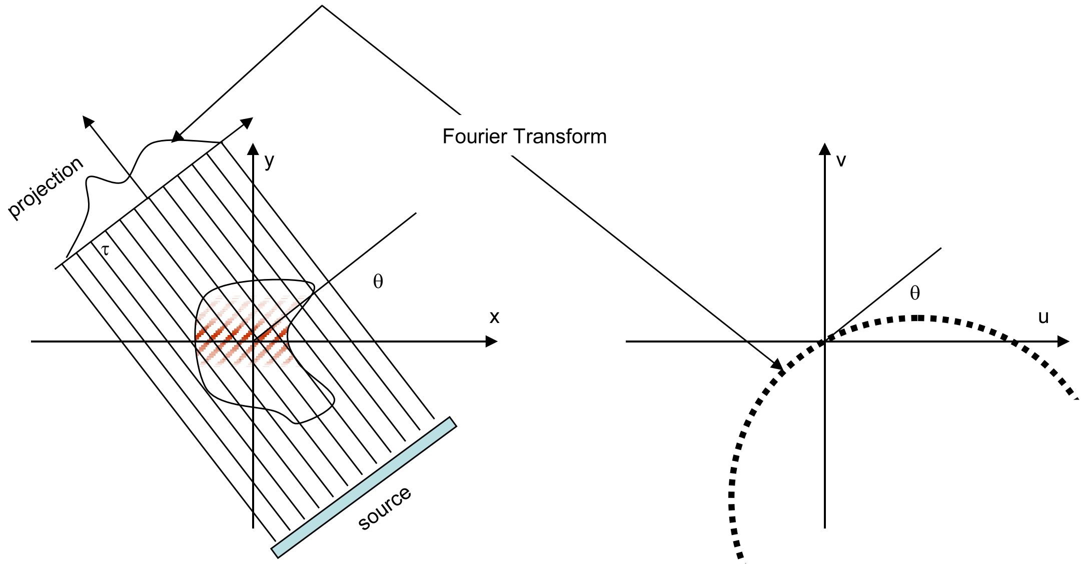


Figure 7.43: Fourier diffraction theory. The Fourier transform of an object's projection forms an arc in 2D k space.

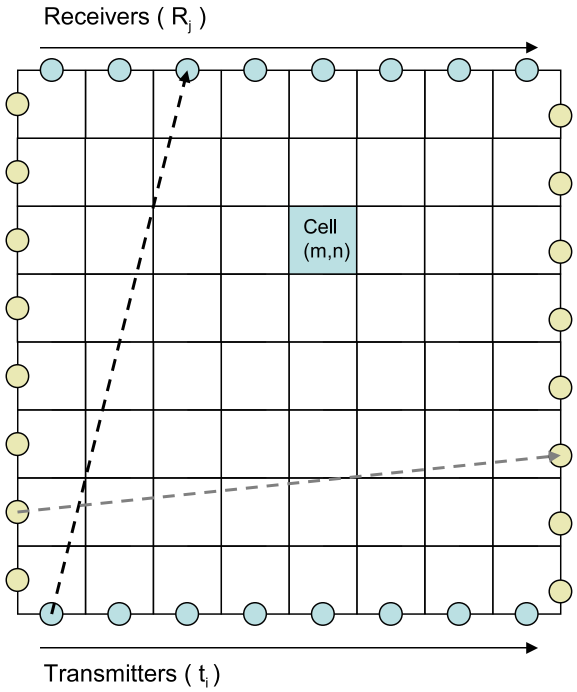


Figure 7.44: Schematic diagram showing double crosshole tomography.

3. Since signal amplitude is nonuniform over the field of view of a transducer, it is not practical to use projections of signal attenuation. Time of flight measurements must be used.

4. It is difficult to measure accurate travel times from a dispersive and possibly multimodal signal.

5. The small variation in time of flight can masked by inaccurate robot positioning, this is discussed further in Section 7.10.5.

As with projection tomography, Double Crosshole Tomography assumes that the ultrasound travels in straight rays, this is a severe approximation for ultrasonic waves, which leads to a loss of fidelity in the resulting image. Malyarenko and Hinders have adapted the algorithm to diffraction tomography [151] by integrating an iterative ray tracing algorithm that can accommodate curved rays. They used a high frequency approximation to the wave equation (rather than Born or Rykov) which is only valid for wavelengths much shorter than characteristic dimensions of the defect or inhomogeneity. Consider the A_0 mode at 600kHz. It has a wavelength of 3.3mm in 1mm thick aluminium, so defects would need to be reasonably large to have characteristics greater than a wavelength.

7.10.5 Positional Inaccuracy and Time of Flight Measurements

Consider a 1mm aluminium sample with dimensions of 150 × 150 mm. A transmitter is placed at (75mm ,0mm) to a receiver at (75mm ,150mm), an ultrasonic signal is transmitted between transducers. There is a 0.5mm thinned region that covers 10% of the path. The excitation is the A_0 Lamb wave mode at 600kHz, so travels at a group velocity of 3000 m/s in the 1mm thick region and 2500 m/s for the thinned region. For an undamaged plate the transit time would equal distance/velocity = 0.15/3000 = $50\mu s$. For the damaged plate, time = distance/velocity = 0.135/3000 + 0.015/2500= $51\mu s$. A delay of 1 μs. 1 μs corresponds to 3mm of travel in undamaged plate.

In this case, a 3mm positional inaccuracy in transducer placement would result in same signal change as a 50% deep, 15mm long thinned defect. The robot positional accuracy was sufficient for SAFT, since the robot only had to drive down a single straight line path (Section 7.5). For tomography, two robots need to travel back and forth to numerous positions, which is more difficult to achieve accurately. With the current hardware (Chapter 3) the robots may have a positional inaccuracy of 15mm each. This positional sensitivity would make time of flight tomography (projection or double cross hole) difficult to implement.

7.11 Length of the Scan Path

All simulated scans so far considered have been artificially small so that simulations could be run on a desktop workstation. However, we will now consider a more realistic 1 × 1 m inspection area, using 20 × 20 mm transducers in one of five possible configurations.

Consider the local pitch-catch configuration from Chapter 6, with a scan interval of 20 mm. The scan path is shown in Figure 7.45 a. The minimum distance travelled by a robot would be $(1000/20) \times 1000 + 1000 = 51000mm = 51m$. This would give a potential resolution of $20 \times 20mm$, assuming the double image [96] was removed in post processing.

Using SAFT or a B-Scan the robot would need to travel (1m) along the bottom edge of the sample, as shown in Figure 7.45 b. The image would have a potential resolution of $3 \times 3.3mm = 10mm$ (length of a 3 cycle wave packet for wide band transducer).

Using the Total Focusing Method, a robot with an ultrasonic transmitter would need to travel 1m. For each transmission, an additional robot with an ultrasonic receiver would need to visit each reception location. This is shown in Figure 7.45 c. If the sample spacing is 1.5mm, the second robot would need to travel 667m. TFM has the same potential resolution as SAFT.

In a projection tomography setup with 72 projections, two robots would each need to travel $72 \times \sqrt(2) \times 1000mm = 101823mm = 101.8m$ (ignoring small distance between projection paths). This is shown in Figure 7.46. The theoretical maximum resolution is $\lambda/2 = 3.3/2 = 1.65mm$, however a value close to this has not yet been demonstrated [152] using ultrasonics.

Consider the Double Crosshole tomography configuration shown in Figure 7.44. In the simplest configuration, a transmitting robot would travel to each transmission location (i= 1 to i = N). For each transmission the robot would need to travel to all possible receiver locations (j = 1 to j = N). This is shown in Figure 7.45

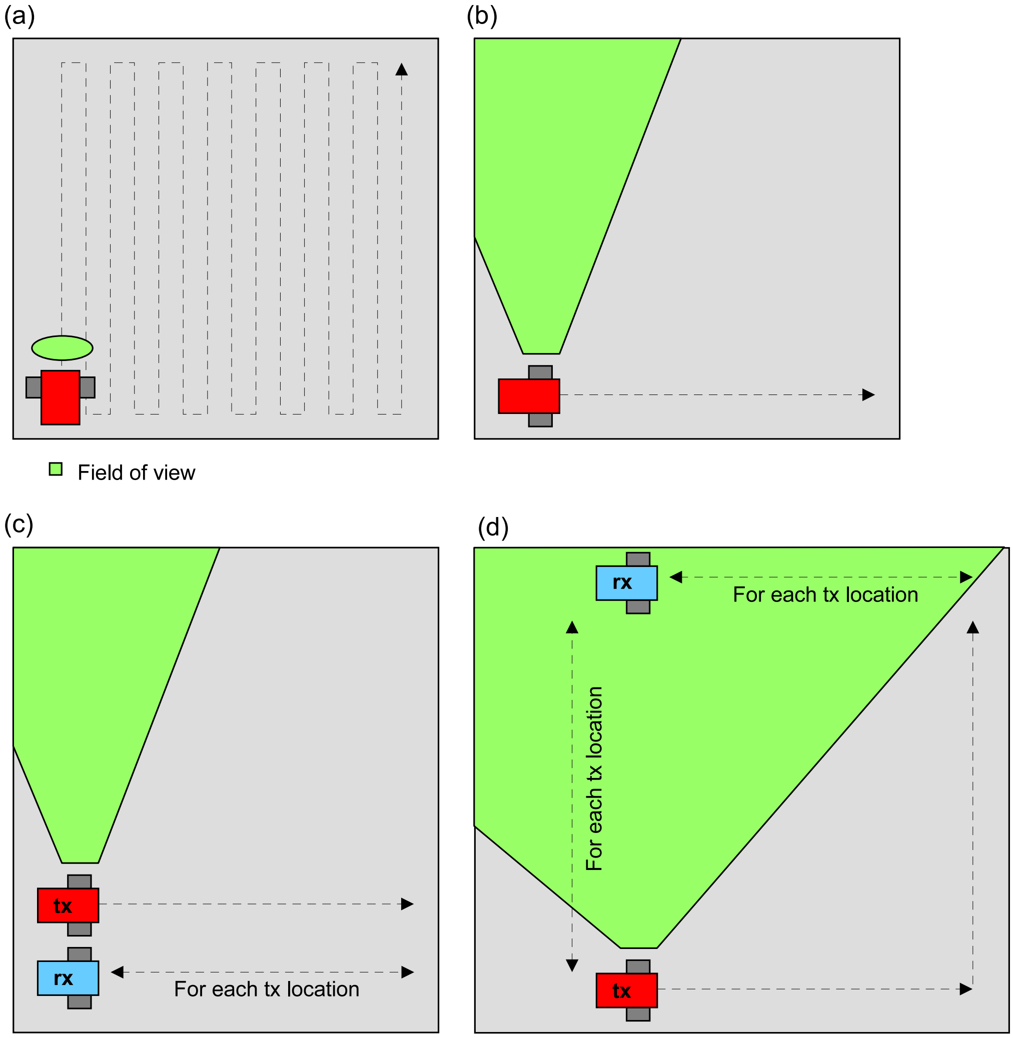


Figure 7.45: Comparison of scan paths I. Tx = transmitting robot, Rx = receiving robot. (a) Local pitch-catch raster, (b) B-scan / SAFT, (c) Full matrix capture & (d) Double cross hole tomography

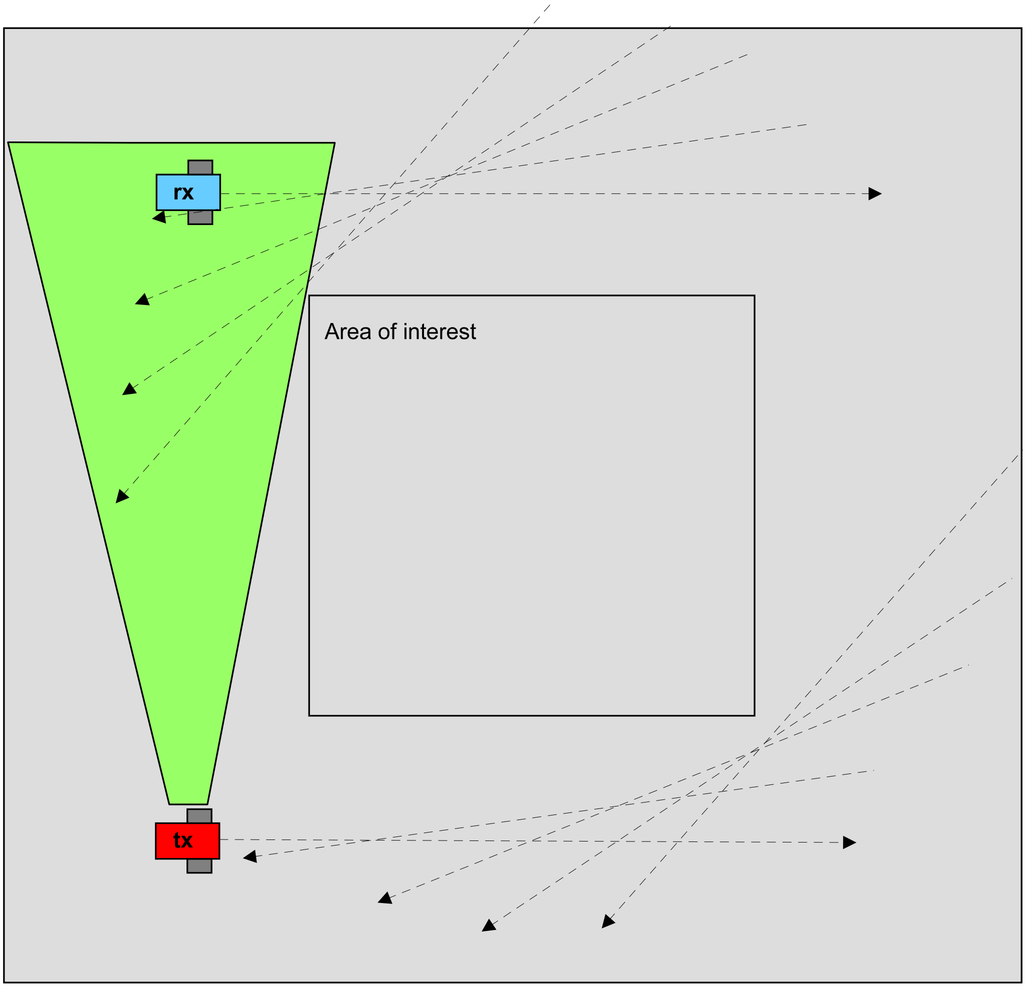


Figure 7.46: Comparison of scan paths II. Tx = transmitting robot, Rx = receiving robot. Parallel beam tomography

d. This imaging strategy requires the transmission robot to travel $2m$ while the receiving robot travels $(1000/1.5) \times 2 * 1000 = 1333 \times 10^3 mm = 1.3km$ This calculation assumes an element spacing of 1.5mm. Double cross-hole tomography has the same theoretical resolutions as diffraction tomography [151] (For 600kHz A_0, $\lambda/2 = 3.3/2 = 1.65mm$).

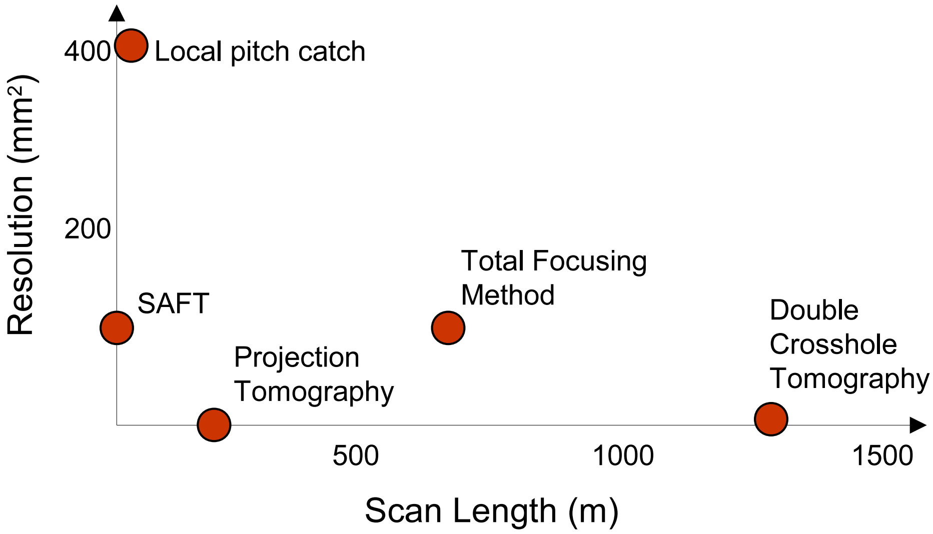


Figure 7.47: The theoretical resolution of different imaging algorithms plotted against robot scan length

Figure 7.47 plots the theoretical resolution of each imaging algorithm plotted against the length of the robot scan path. It is clear from these results that tomographic imaging algorithms are impractical for a scanner based on miniature robotic vehicles due to the excessively long scan paths.

7.12 Reconfigurable Scanning

7.12.1 Finding Plate Boundary

As discussed in Chapter 4, the Lamb wave inspection payload works best on plate specimens. In real world structures, the plates are riveted to a support structure or welded together. In such cases, the structure is comprised of a series of plates, with boundaries that severely attenuate Lamb wave propagation. In order to intelligently scan such a structure, the layout of the panels must be known. Since the required level of detail is often unavailable, even with engineering drawings, it would be beneficial if an RSA could both map the structure and locate defects relative to this map. This Section attempts to combine positioning and context awareness with pulse-echo scanning using a single robot to map its location relative to the nearest plate boundaries.

The robot uses transducers configured for pulse-echo (See Figure 6.1, Chapter 6). The transducers are fixed at the appropriate angle to generate and receive the zeroth order anti-symmetric (A_0) Lamb wave in the specimen.

The algorithm assumes that the structure is made up of rectangular plate sections, of unknown dimensions or orientation. The RSA drives onto the plate section and performs a rotational pulse-echo scan which picks up reflections from the plate edges, corners and some defects. An example rotational scan, in simulation, of the 1mm aluminium plate shown in Figure 7.48 is shown in Figure 7.49. The virtual RSA was positioned at (100,100) and performed 72 measurements at 5^o intervals. The plate contained two defects, both of 100% depth.

Reflections from the edges and corners can be clearly seen. There are also faint reflections from the defects. The algorithm extracts the peaks and converts them to distances using the algorithm discussed in Section 7.6. These are then converted to Cartesian coordinates to produce Figure 7.50. The robot now moves to a second location, in this case x = 170 mm, y = 170 mm, and repeats the rotational scan. The peaks are again extracted and added to the points from the first scan. The algorithm then fits a rectangle to the total collection of coordinates. This reveals the plate boundary shown in Figure 7.53.

The rectangle is characterised by five variables as shown in Figure 7.51; the location of the bottom left corner (x, y), the width 'W', the length 'L' and the angle of rotation from the normal 'θ'. The distance between each point and this rectangle

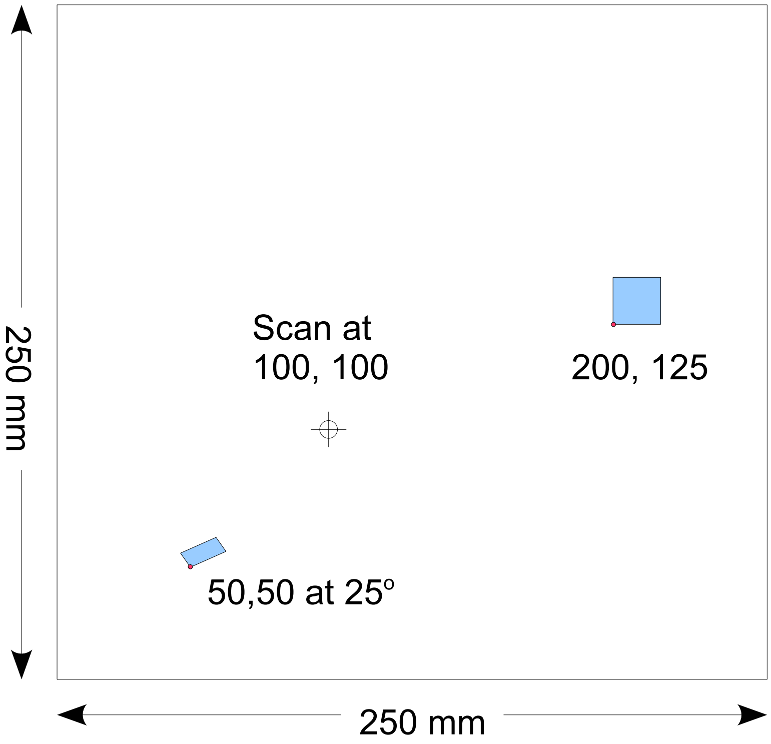


Figure 7.48: Schematic of example plate

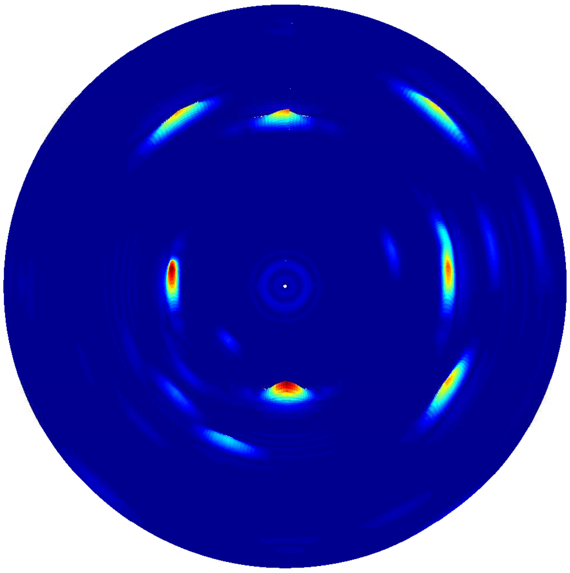


Figure 7.49: Simulation of rotational pulse-echo of 250mm x 250mm plate section

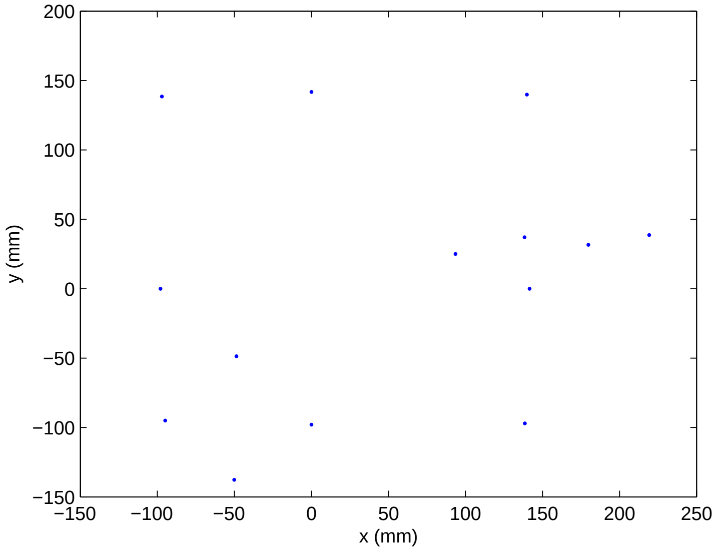


Figure 7.50: Extraction of peaks from Figure 7.49

(Figure 7.52) can be calculated by calculating the shortest distance between the point and the four line segments. The quality of the fit can be defined by the mean squared error (MSE). Once the problem has been defined, standard optimisation techniques can be applied to solve for the 'best-fit' rectangle. In this case Matlab's 'fminunc' function was used. [73].

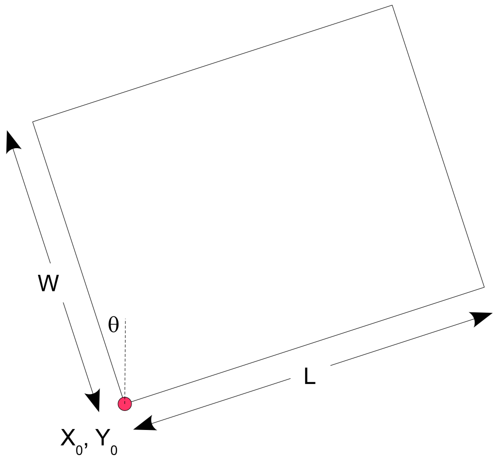


Figure 7.51: Properties of the best-fit rectangle

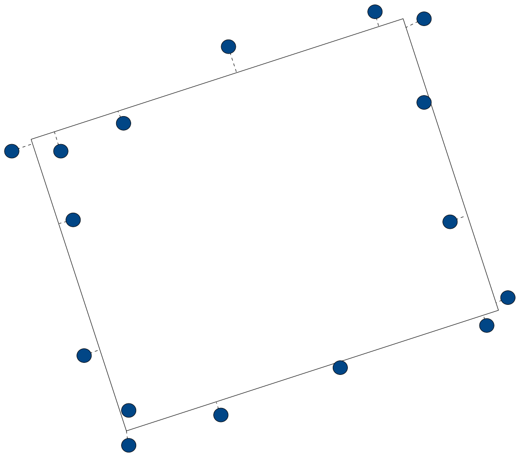


Figure 7.52: Distance between each point and the best-fit rectangle

In the current form, the algorithm is sensitive to spurious reflections from defects. To overcome this limitation, a threshold was set for a maximum 'average point deviation' from the rectangle (i.e. 5mm) and if the optimisation solution fell above this threshold the worst offending point was removed. This process was repeated until the error fell below the maximum deviation threshold. An example fit is shown in Figure 7.53. In this case seven points have been removed before a satisfactory fit was achieved (removed points are shown in blue). The resulting plate was within 5 mm for x_0, y_0, L and W and 0.6^o for θ of the correct value.

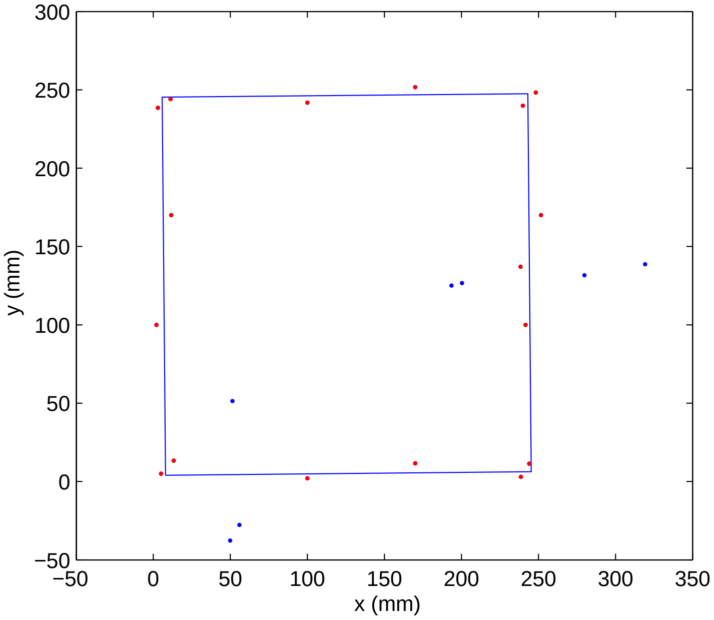


Figure 7.53: Resulting plate boundaries in the presence of spurious reflections from defects

Figure 7.54 shows the results from an experimental pulse-echo scan of a 300 × 300 mm, 1 mm plate specimen. The peaks from the four edges can be clearly seen. However, some of the weak reflections from the corners have been obscured by noise and spurious reflections. Figure 7.55 shows the extracted peaks. This data was combined with the peaks from an further rotational scan to produce Figure 7.55. The estimated plate size was 295 × 304 mm with an angle of 3.8^o, which is reasonably close to the actual values of 300 × 300 mm at 0^o. Reducing the angular step size from 5^o would improve the accuracy of the algorithm.

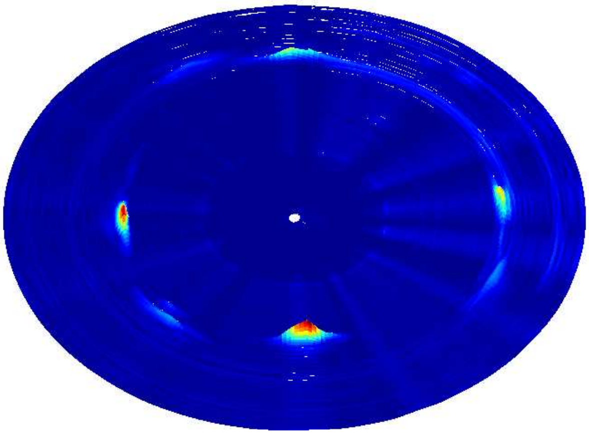


Figure 7.54: Experimental rotational pulse-echo of scan of 300mm x 300mm plate section

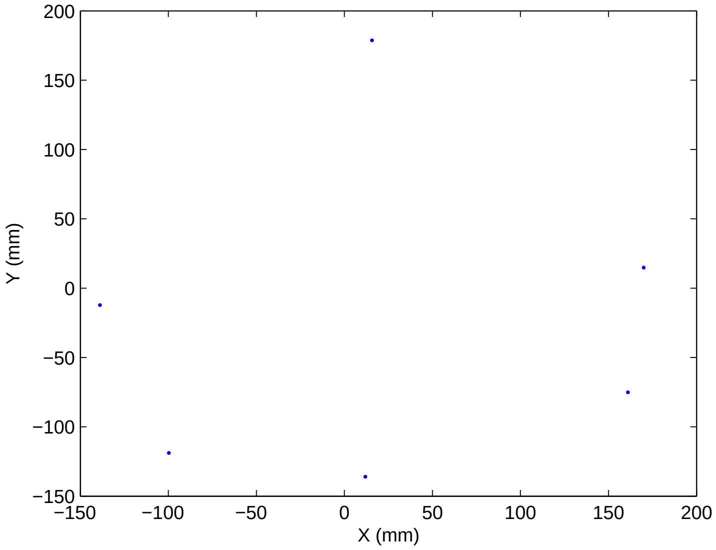


Figure 7.55: Extraction of peaks from Figure 7.54

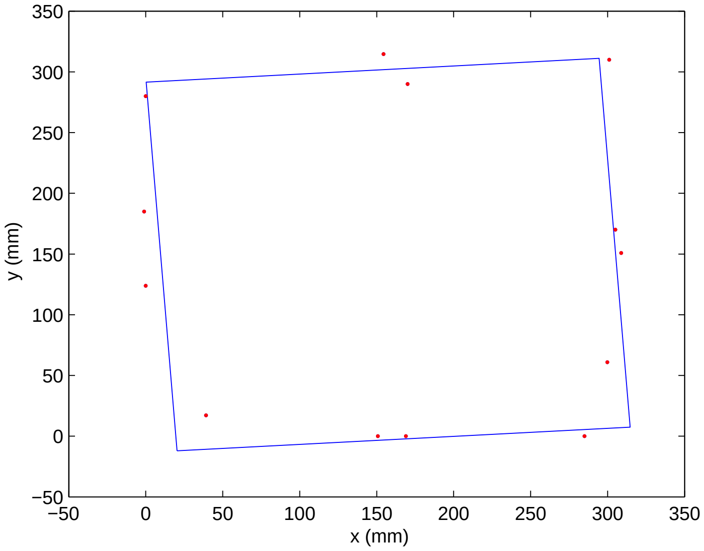


Figure 7.56: Experimental plate boundaries fitted to experimental data. The data was taken Figure 7.55 and an additional rotational scan

7.12.2 B-Scan

Since defects may reflect the ultrasound away from the source, B-scans are not a dependable way of identifying defects. However, if the B-scan is aligned with the plate boundary, it is then possible to detect defects, not by their reflection, but by the shadow they cast over the boundary. In the previous section, it was shown that the plate boundary can be estimated from two rotational pulse-echo scans. A more robust NDE inspection would first estimate the plate boundary, then align B-scans parallel to the plate edges. This is demonstrated in the following example.

Continuing the inspection started in Section 7.12.1, the RSA was then instructed to perform two B-scans, moving horizontally across the plate at y = 180 mm. In the first scan, the robot was orientated to transmit ultrasound toward the top boundary (y = 250mm). In the second scan, the robot transmitted ultrasound toward the bottom boundary (y = 0mm). The results are shown in Figures 7.57 and 7.58.

Figure 7.57 clearly recreates the plate edge, showing there are no defects. Figure 7.58 shows two distinct shadows in the back wall, one at x = 55mm and another at x = 210mm. These correspond with the two defects. The defect at 200, 200 mm is also clearly visible, whereas the defect at 50,50 mm is very faint. This is because the angle of the edge reflects the ultrasonic wave away from the RSA.

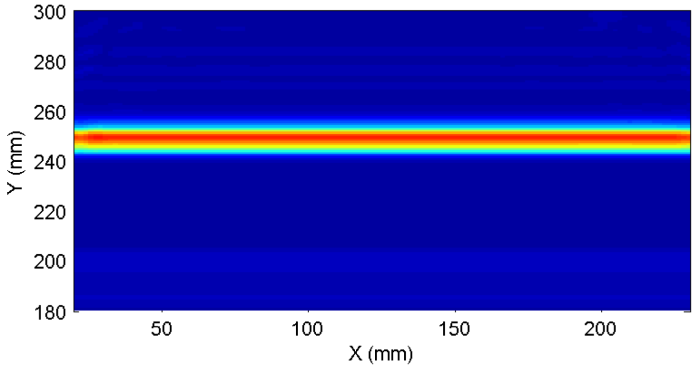


Figure 7.57: B-scan example, imaging y > 180 mm. See Figure 7.48 for plate geometry.

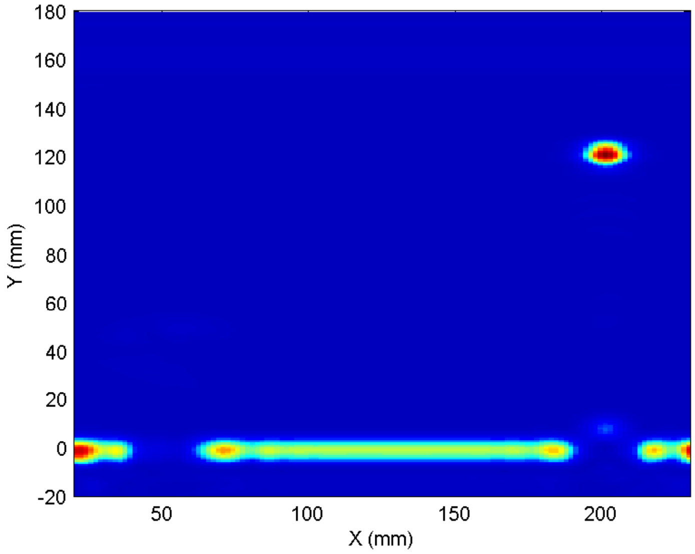


Figure 7.58: B-scan example, imaging y < 180 mm. See Figure 7.48 for plate geometry.

7.13 Conclusions and Future Work

This chapter has evaluated several imaging algorithms that make use of long range Lamb wave propagation. It began with a detailed evaluation of the Synthetic Aperture Focusing Technique (SAFT), which is often used to enhance B-scan results. The SAFT algorithm requires a wide beam divergence from the transducer so that the same reflection is visible in multiple receiver locations. It was found that the 30 × 30 mm transducers used in the ultrasonic payload discussed in Chapter 4, had a very narrow beam with negligible divergence. In such a case, SAFT offered no improvement over standard B-scans. Simulation, using the software created in Chapter 5, showed that in order to achieve a reasonable beam divergence, of for example 15^o, the transducer dimensions would need to be reduced to 8 × 10 mm. Constructing these transducers fell outwith the scope of the current project, but has been highlighted as a starting point for future work.

The simulation results should be verified experimentally. One possible method of verification would be to use a laser interferometer to record plate displacement under excitation for varies geometries of transducer. This was attempted, but the available Polytec OFV interferometer [153] lacked the sensitivity required to measure a Lamb wave generated from an angled air coupled transducer.

Several tomographic algorithms, including straight ray and projection tomography and double crosshole tomography were evaluated and compared with B-Scanning. It was concluded that the robot scan paths would be too long for practical implementation. For example, to scan a 1 × 1 m sample using projection tomography, two robots would need to travel over 100m each. In addition to this, techniques that require time of flight measurement are extremely sensitive to the uncertainty in robot positioning, prevalent in a real world system. This lead to the conclusion that B-Scanning, potentially enhanced by SAFT depending on the transducer geometry, was the most suitable imaging technique for use on the miniature robotic vehicle platform.

In order to automatically scan a plate specimen, the plate boundaries need to be determined. The chapter concluded with an algorithm to automatically resolve plate boundaries for two or more rotational pulse-echo scans. This algorithm facilitated intelligent orientation of scans for in-situ inspection. The algorithm was found to be robust to spurious reflections from defects and the low level of noise evident in experimental results.

Chapter 8

Conclusions and Future Work

8.1 General Overview

A novel reconfigurable ultrasonic inspection system has been presented. The scanner comprises of a fleet of miniature robotic vehicles, each carrying a non-contact ultrasonic payload, capable of generating and measuring the A_0 Lamb wave mode. The individual robots have a volume of approximately 150 × 150 × 100 mm and weigh 610g. They feature an embedded Linux computer for extensive on-board processing capabilities and high bandwidth communications over WiFi. A Lithium Polymer battery facilitates four hours of continuous operation. The robots use magnetic wheels to adhere to ferromagnetic structures, facilitating motion over three dimensional specimens. The robots are accurately positioned using an extended Kalman filter.

A computer simulation tool was created to facilitate the effective design and evaluation of ultrasonic imaging algorithms. Transducer behaviour was simulated using a linear systems (LS) approximation. Wave propagation in the structure was modelled using the Local Interaction Simulation Approach (LISA). Integration of the LS and LISA approaches were validated for use in Lamb wave scanning by comparison with both analytical techniques and more computationally intensive commercial finite element/difference codes. Starting with fundamental dispersion data, the work proceed to describe the simulation of wave propagation and the subsequent interaction with artificial defects and plate boundaries.

Two inspection approaches were considered. The first approach centred upon a pair of angled pitch-catch transducers over a small area under the robot. A Lamb wave was generated in the specimen, then recorded by the receiving transducer. A change in material thickness, or an inhomogeneity that caused scattering, reduced the receiver amplitude thus indicating a defective region. An algorithm was created that pieced together the series of measurements to create a map of the specimen. The second approach configured the transducers in pulse-echo mode. In this configuration, the Lamb wave was propagated forward and the receiver listened for echos from reflectors. Reflections were typically emitted from defective regions or plate boundaries. By taking a sequence of these measurements it was possible to build up a map of the structure (B-scan). This was also demonstrated experimentally. The work then attempted to enhance the B-scan results using the Synthetic Aperture Focusing Technique (SAFT).

SAFT requires a reasonably wide beam divergence from the ultrasonic transducer. However, simulation revealed that the beam divergence of ultrasonic transducers is negligible for diameters greater than 15 mm. Simulations of SAFT with 3mm wide transducers demonstrated the effectiveness of SAFT if wide beam divergence can be achieved. Simulations also showed that SAFT was very sensitive to a positional inaccuracy of the robotic vehicles, leading to a distortion of results. The positional sensitivity could be reduced by limiting the transducer's field of view, but at the cost of reduced image fidelity.

Several tomography algorithms were evaluated, but it was concluded that the large number of measurements required made them unsuitable for use on the robotic platform.

Finally, an algorithm was developed that utilised Lamb wave ranging measurements to position the robot relative to a plate sample. The algorithm took the following form; firstly, the robot was instructed to take pulse-echo Lamb wave measurements at a series of angles covering 360^o. The peaks were extracted and converted to Cartesian coordinates. A 'best-fit' algorithm was used to fit a rectangular plate, of unknown dimensions or orientation, to the points (ignoring outliers). The robot's position was calculated relative to the best-fit rectangle. The algorithm performed robustly, even in the presence of spurious reflections from defects, or the low level of noise found in experimental results.

8.2 Suggestions for Further Work

8.2.1 Short Term Refinements

The most significant limitation of the current system is the relatively large 30 × 30 mm transducers used in the ultrasonic payload (Chapter 4). In Chapter 6 a short range pitch-catch imaging algorithm was presented. The image fidelity was directly related to the transducer size. Reducing the size would improve image quality. In Chapter 7 SAFT was evaluated. It was found that SAFT required significant beam divergence to be effective. For example,in order to generate a reasonable beam divergence of 15^o, the air-coupled transducers should have dimensions of less than 8 × 10 mm. Future work will attempt to reduce the size of the transducers while maintaining sufficient SNR.

Switching from air-coupled transducers to contact probes has the potential to dramatically improve SNR, but would require either a wheel probe or a reservoir of coupling fluid for liquid coupling. However, wheel probes do not tend to couple well to the dirty unpolished surfaces typically found in industry and the requirement for a reservoir of coupling fluid, limits the number of possible measurements in a single scan.

If adequate coupling could be achieved, small angled transducers could be used to generate transducers with sufficient filed of view and SNR. Additionally, this would allow the excitation of the S_0 Lamb wave mode, rather than A_0, which has less out-of-plane displacement and hence less attenuation. $S - 0$ would facilitate longer range inspections. Alternatively, the transducer could be positioned parallel to the surface, exciting an omnidirectional wave. This would increase the scope for reconfigurable imaging algorithms, but does not allow Lamb wave mode selection. This makes it much more difficult to interpret results which contain several Lamb wave modes. In addition to this, boundary reflections are much harder to identify from omnidirectional transducers. Future work should first evaluate these configurations in simulation to evaluate the potential improvement in ultrasonic imaging, before proceeding to address the coupling problem.

Accurate robot positioning will form a core component of future research. The Kalman filter study in Chapter 3 has set a benchmark of positional accuracy using the current hardware. In the near future, the Kalman filter should be compared with the Particle filter, both in simulation and experimentation. This comparison will determine if the Particle filter warrants the additional computation for this particular application. In addition to this, the relatively low accuracy Cricket Indoor Positioning System will be replaced with a more accurate system; ideally the ultrasonic phase based ranging system being developed in the Centre for Ultrasonic Engineering.

The recently purchased Vicon MX positioning system, with sub-millimetre accuracy, will fulfil two roles. Firstly, it will be used to provide accurate positioning data directly to the robots, obviating the positioning problem and thus allowing research to focus elsewhere. Secondly, it will be used as a benchmarking tool that can compare both algorithms and new positioning systems. This will replace the in-house visual positioning system discussed in Chapter 2.

There is scope to improve the mechanics of the miniature robotic vehicles discussed in Chapter 2. The design lends itself to further miniaturisation, thus allowing improved access to confined spaces. In addition to this, effort should be made to improve the robots mobility over real world structures. The differential drive mechanics may be replaced, or complemented with mechanisms to negotiate obstacles, such as flanges or 'H' beams. Furthermore, the design could be adapted for underwater use, opening up a range of applications including the in-situ inspection of ships.

The current robot's traction is designed for ferromagnetic materials, such as steel. Alternative traction methodologies would expand the scope of the system to non-ferromagnetic structures, facilitating inspections in the aviation industry. There are numerous alternative traction technologies, as discussed in Section 2.1. However, these alternatives are not yet mature enough for deployment in an industrial setting. None of these technologies offer the robustness, simplicity, and scope for miniaturisation that permanent magnets provide. Many of the technologies require a constant supply of power, reducing wireless operating time and reducing reliability (an empty battery or fault will cause the robot to lose traction). That said, the most promising approach is the vacuum based vortex technology used by Clarifying Robotics [34]. Future work will adapt this technology for use in the robotic platform.

8.2.2 Longer Term Vision

It is generally inconvenient, or even impossible to construct a global positioning framework for in-situ inspections. A key feature of the miniature robotic vehicles is their ability to access difficult to reach or dangerous locations. This capability is severely reduced if the robots need a preassembled global positioning framework. The Simultaneous Localisation and Mapping Algorithm (SLAM) [11] facilitates accurate robot positioning without the need for a global positioning framework. The algorithm extracts positional information from the environment, continually constructing and refining an internal map. The internal map is then used for localisation. SLAM is best suited to feature rich environments, such as industrial plant, which by design have a high density of features like pipes, supports and tanks. These settings offer the ideal setting for SLAM. SLAM refines the map when it detects the same feature more than once. This makes it ideally suited to robots performing raster scans, since the same features will be in view numerous times. Future work will integrate SLAM into the miniature robotic vehicle platform.

In the future, the robot fleet will become entirely autonomous. An operator will deploy the robotic fleet and wait while the robots intelligently inspect the structure. The robots would work together, dynamically optimising the inspection based on incoming results. For example, an ultrasonic robot may rapidly map the plate sections using the mapping algorithm discussed at the end of Chapter 7. It would then screen the plates using the B-scan algorithm discussed in Chapter 7. Additional robots performing the local pitch-catch algorithm discussed in Chapter 6, eddy current or MFL inspections [8], would inspect areas that were identified by the ultrasonic screening algorithm. Such an approach would require reasonable context awareness, either from a predefined CAD drawing of the structure or SLAM generated map. Automatic defect classification algorithms would be required to distinguish between structural features, such as rivets, and defective regions.

The miniature robotic vehicle work will be driven forward on two fronts. Firstly, research will continue, focusing primarily on robot positioning, the integration of SLAM and the creation of a unified Baysian framework to incorporate robot positioning and the NDE payloads. This is supported for an additional three years as a core RCNDE [7] project. In addition to this, a commercial operation will look to customise the current platform for industrial applications. This being supported by the Royal Society of Edinburgh Enterprise Fellowship. The first commercial application is focussed on the measurement of wall thickness of a tank inside a nuclear processing cell. This measurement will facilitate an accurate projection of the tank's lifetime, something that is critical for the ongoing use of the facility. The readiness of both academia and industry to continually support the work is an excellent endorsement of the work to date and its future potential.

Bibliography

[1] L. Feng, HR Everett, and J. Borenstein. *Where Am I?: Sensors and Methods for Autonomous Mobile Robot Positioning.* Dept. of Mechanical Engineering and Applied Mechanics, University of Michigan, 1994.

[2] Araldite, produced by Huntsman. http://www.huntsman.com/, Accessed Sept 2009.

[3] Morgan Electroceramics. http://www.morgantechnicalceramics.com, Accessed Dec 2008.

[4] Technical Insights: a division of consulting firm Frost & Sullivan. Nondestructive Testing. New Developments and Growth Opportunities, 2003.

[5] JM Farley, M. Babcock, and UK Renfrew. Best Practice in the Application of NDT – An Update. In *World Conference on NDT*, 2004.

[6] R. Singh. Three decades of NDI reliability assessment. *Contract*, 41608(99-C):404, 2000.

[7] Research Centre for Non Destructive Evaluation. http://www.rcnde.co.uk/, Accessed Sept 2009.

[8] Markus Friedrich. *Design of Miniature Mobile Robots for Non-Destructive Evaluation.* PhD thesis, University of Strathclyde, 2007.

[9] I. A. Viktorov. *Rayleigh and Lamb waves: Physical theory and applications.* Plenum Press, 1967.

[10] R. Farlow and G. Hayward. An automated ultrasonic NDT scanner employing advanced air-coupled 1-3 connectivity composite transducers. *Insight*, 38(1):41–50, 1996.

[11] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 128(1-2):99–141, 2001.

[12] R.E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[13] R. Lastra, P. Vallejos, and J. Ruiz-del Solar. Self-localization and ball tracking for the robocup 4-legged league. In *Proceeding of the 2nd IEEE Latin American Robotics Symposium LARS*, 2005.

[14] S. Shoval, I. Zeitoun, and E. Lenz. Implementation of a Kalman filter in positioning for autonomous vehicles, and its sensitivity to the process parameters. *The International Journal of Advanced Manufacturing Technology*, 13(10):738–746, 1997.

[15] SI Roumeliotis and GA Bekey. Collective localization: a distributed Kalman filter approach tolocalization of groups of mobile robots. In *IEEE International Conference on Robotics and Automation, 2000. Proceedings. ICRA'00*, volume 3, 2000.

[16] R. Sicard, A. Chahbaz, and J. Goyette. Guided Lamb waves and L-SAFT processing technique for enhanced detection and imaging of corrosion defects in plates with small depth-to wavelength ratio. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 51(10):1287–1297, 2004.

[17] N. Correll and A. Martinoli. Collective inspection of regular structures using a swarm of miniature robots. page 375, 2006.

[18] A. Correll, N. Martinoli. Multirobot inspection of industrial machinery. *Robotics & Automation Magazine, IEEE*, 16:103–112, 2009.

[19] L.L. Menegaldo, M. Santos, G.A.N. Ferreira, R.G. Siqueira, and L. Moscato. SIRUS: A mobile robot for Floating Production Storage and Offloading (FPSO) ship hull inspection. In *10th IEEE International Workshop on Advanced Motion Control, 2008. AMC'08*, pages 27–32, 2008.

[20] AA Carvalho, IC Silva, JMA Rebello, RO Carneval, and JAB Farias. Inspection of ship hulls using automated ultrasonic inspection. *Insight-Non-Destructive Testing and Condition Monitoring*, 47(12):744–747, 2005.

[21] T. Sogi, Y. Kawaguchi, H. Morisaki, K. Ohkawa, N. Kai, and H. Hayakawa. Inspection robot for spherical storage tanks. *Industrial Electronics Society, 2000. IECON 2000. 26th Annual Confjerence of the IEEE*, 1, 2000.

[22] S. Burkett and H. Schempf. Wireless Self-powered Visual and NDE Robotic Inspection System for Live Gas Distribution Mains. Technical report, Carnegie Mellon University, 2006.

[23] J. Shang, T. Sattar, S. Chen, and B. Bridge. Design of a climbing robot for inspecting aircraft wings and fuselage. *Industrial Robot: An International Journal*, 34(6):495–502, 2007.

[24] Z. Pan and Z. Zhu. Miniature pipe robots. *Industrial Robot: An International Journal*, 30(6):575–583, 2003.

[25] LD Mackenzie, SG Pierce, and G. Hayward. Robotic Inspection System For Non-Destructive Evaluation (NDE) Of Pipes. In *AIP Conference Proceedings*, volume 1096, page 1687, 2009.

[26] SilverWing. http://www.silverwinguk.com/, Accessed Sept 2009.

[27] W. Fischer, F. Tâche, R. Siegwart, A. Charoy, and A. Mange. Magnetic wall climbing robot for thin surfaces with specific obstacles. In *Field and Service Robotics: Results of the 6 th International Conference(STAR: Springer Tracts in Advanced Robotics Series Volume 42)*, volume 42, pages 551–561. Springer, 2008.

[28] PG Backes, Y. Bar-Cohen, and B. Joffe. The multifunction automated crawling system (MACS). In *1997 IEEE International Conference on Robotics and Automation, 1997. Proceedings.*, volume 1, 1997.

[29] JC Grieco, M. Prieto, M. Armada, and P. Gonzalez de Santos. A six-legged climbing robot for high payloads. In *Control Applications, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 1, 1998.

[30] BL Luk, KP Liu, AA Collie, DS Cooke, and S. Chen. Tele-operated climbing and mobile service robots for remote inspection and maintenance in nuclear industry. *Industrial Robot: An International Journal*, 33(3):194–204, 2006.

[31] H. Prahlad, R. Pelrine, S. Stanford, J. Marlow, and R. Kornbluh. Electroadhesive robotswall climbing robots enabled by a novel, robust, and electrically controllable adhesion technology. In *IEEE International Conference on Robotics and Automation, 2008. ICRA 2008*, pages 3028–3033, 2008.

[32] C. Menon, M. Murphy, and M. Sitti. Gecko inspired surface climbing robots. In *IEEE International Conference on Robotics and Biomimetics, 2004. ROBIO 2004*, pages 431–436, 2004.

[33] M. Sitti and RS Fearing. Synthetic gecko foot-hair micro/nano-structures for future wall-climbing robots. In *IEEE International Conference on Robotics and Automation, 2003. Proceedings. ICRA'03*, volume 1, 2003.

[34] Clarifying Technologies . http://www.clarifyingtech.com, Accessed Sept 2009.

[35] David Reinfeld Lewis Illingworth. Suction cup vortex attractor, 2005.

[36] J. Xiao, A. Calle, A. Sadegh, and M. Elliott. Modular wall climbing robots with transition capability. In *2005 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 246–250, 2005.

[37] G. Caprari and R. Siegwart. Design and control of the mobile micro robot alice. pages 23–32, 2001.

[38] *Fascination of down scaling - Alice the sugar cube robot*, volume 1. Springer, 2001.

[39] T. Sashida and T. Kenjo. *An introduction to ultrasonic motors*. Oxford University Press, USA, 1993.

[40] BR Donald, CG Levey, CD McGray, I. Paprotny, and D. Rus. An untethered, electrostatic, globally controllable MEMS micro-robot. *IEEE Journal of Microelectromechanical Systems*, 15(1):1–15, 2006.

[41] B. Wilcox, A. Nasif, and R. Welch. Implications of Martian Rock Distributions on Rover Scaling. In *Planetary Society International Conference on Mobile Planetary Robots and Rover Roundup*, 1997.

[42] A F T Winfield and Julien Nembrini. Safety in numbers: Fault tolerance in robot swarms. *International Journal on Modelling Identification and Control*, 1(1):30–37, 2006.

[43] R. Grabowski, L.E. Navarro-Serment, C.J.J. Paredis, and P.K. Khosla. Heterogeneous teams of modular robots for mapping and exploration. *Autonomous Robots*, 8(3):293–308, 2000.

[44] L.E. Navarro-Serment, C.J.J. Paredis, P.K. Khosla, et al. A beacon system for the localization of distributed robotic teams. In *Proceedings of the International Conference on Field and Service Robotics*, volume 6. Citeseer, 1999.

[45] K-Team. http://www.k-team.com/, Accessed Sept 2009.

[46] G. Antonelli, S. Chiaverini, and G. Fusco. A calibration method for odometry of mobile robots based on the least-squares technique: Theory and experimental validation. *IEEE Transactions on Robotics*, 21(5):994–1004, 2005.

[47] G. Antonelli, F. Arrichiello, and S. Chiaverini. Experiments of formation control with collisions avoidance using the Null-Space-based Behavioral control. 2006.

[48] G. Antonelli, F. Arrichiello, and S. Chiaverini. Flocking for Multi-Robot Systems via the Null-Space-based Behavioral Control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008. IROS 2008*, pages 1409–1414, 2008.

[49] G. Antonelli, F. Arrichiello, and S. Chiaverini. The Entrapment/Escorting Mission. *IEEE Robotics & Automation Magazine*, 15(1):22–29, 2008.

[50] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm intelligence: from natural to artificial systems.* Oxford University Press, USA, 1999.

[51] F. Mondada, LM Gambardella, D. Floreano, S. Nolfi, J.L. Deneuborg, and M. Dorigo. The cooperation of swarm-bots: Physical interactions in collective robotics. *IEEE Robotics & Automation Magazine*, 12(2):21–28, 2005.

[52] F. Mondada, G. C. Pettinaro, A. Guignard, I. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L.M. Gambardella, and M. Dorigo. Swarm-bot: a new distributed robotic concept. *Autonomous Robots*, 17(2–3):193–221, 2004.

[53] M. Dorigo R. Gro. Evolution of solitary and group transport behaviors for autonomous robots capable of self-assembling. *Adaptive Behavior*, 16(5):285–305, October 2008.

[54] M. Dorigo A.L. Christensen, R. O'Grady. Morphology control in a multirobot system. *IEEE Robotics & Automation Magazine*, 11(6):732–742, 2007.

[55] Swarmanoid Project - http://www.swarmanoid.org (Checked 29/09/09).

[56] HB Brown Jr, J.M.V. Weghe, CA Bererton, and PK Khosla. Millibot trains for enhanced mobility. *IEEE/ASME transactions on mechatronics*, 7(4):452–461, 2002.

[57] E. Şahin and A. Winfield. Special issue on swarm robotics. *Swarm Intelligence*, 2(2):69–72, 2008.

[58] H. Durrant-Whyte and T. Bailey. Simultaneous localisation and mapping (slam): Part i the essential algorithms. *Robotics and Automation Magazine*, 13:99–110, 2006.

[59] Paul Newman. *On the structure and solution of the simultaneous localization and mapping problem.* PhD thesis, University of Sydney, Sydney, Australia, 1999.

[60] Michael Bosse, Paul M. Newman, John J. Leonard, and Seth Teller. SLAM in Large-scale Cyclic Environments using the Atlas Framework. *The International Journal of Robotics Research*, 23(12):1113–1139, 2004.

[61] S. Thrun and M. Montemerlo. The GraphSLAM algorithm with applications to large-scale mapping of urban structures. *International Journal on Robotics Research*, 25(5/6):403–430, 2005.

[62] S. Thrun, Y. Liu, D. Koller, A.Y. Ng, Z. Ghahramani, and H. Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. *International Journal of Robotics Research*, 2004.

[63] M. Bryson and S. Sukkarieh. Architectures for Cooperative Airborne Simultaneous Localisation and Mapping. *Journal of Intelligent and Robotic Systems*, 55(4):267–297, 2009.

[64] Objet. http://www.objet.com, Accessed Sept 2009.

[65] Gumstix. www.gumstix.com, Accessed Sept 2009.

[66] Analog Devices. *ADXRS150: 150/s Single Chip Yaw Rate Gyro with Signal Conditioning*, rev b edition, 2004.

[67] Texus Instruments. *L293, L293D: Quadruple Half-H Drivers*, rev nov 2004 edition, November 2004.

[68] P. Horowitz and W. Hill. *The art of electronics.* Cambridge Univ Pr, 1989.

[69] National Semiconductor. *LM2825: Simple Switcher 1A DC-DC*, rev 5 edition, 1997.

[70] I. Buchmann and Cadex Electronics Inc. *Batteries in a Portable World: A Handbook on Rechargeable Batteries for Non-Engineers.* Cadex Electronics, 1997.

[71] Microsoft. .NET. http://www.microsoft.com/NET/, Accessed Sept 2009.

[72] Microsoft. DirectX. http://www.microsoft.com/games/en-US/aboutgfw/Pages/directx10.aspx, Accessed Sept 2009.

[73] MathWorks. Matlab. www.mathworks.co.uk, Accessed Sept 2009.

[74] N.B. Priyantha. *The Cricket Indoor Location System.* PhD thesis, Massachusetts Institute of Technology, 2005.

[75] A. Smith, H. Balakrishnan, M. Goraczko, and N. Priyantha. Tracking moving devices with the cricket location system. *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, pages 190–202, 2004.

[76] F.A. Everest and N.A. Shaw. Master handbook of acoustics. *The Journal of the Acoustical Society of America*, 110:1714, 2001.

[77] Vicon . http://www.vicon.com, Accessed Sept 2009.

[78] COMedia. http://www.comedia.com.hk/, Accessed Sept 2009.

[79] A. Kenyon, A. Millar, P. Murray, O. Sadiq, R. Summan, and R. West. Biologically inspired guidance of robots. Master's thesis, University Of Strathclyde, 2008.

[80] J. Borenstein and L. Feng. Measurement and correction of systematic odometry errors in mobilerobots. *Robotics and Automation, IEEE Transactions on*, 12(6):869–880, 1996.

[81] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics.* MIT Press, 2005.

[82] J. Borenstein, HR Everett, L. Feng, and D. Wehe. Mobile robot positioning: Sensors and techniques. *Journal of Robotic Systems*, 14:231–249, 1997.

[83] Analog Devices. http://www.analog.com/, Accessed Sept 2009.

[84] Borenstein, J. Umbmark: a method for measuring, comparing, and correcting dead-reckoning errors in mobile robots. Technical report, Dept. of Mechanical Engineering and Applied Mechanics, University of Michigan, 1994.

[85] Us digital.

[86] Jeffrey Hightower and Gaetano Borriello. A survey and taxonomy of location systems for ubiquitous computing. Technical report, University of Washington, Computer Science and Engineering, 2001.

[87] R. Negenborn. Robot localization and kalman filters. Master's thesis, Utrecht University, 2003.

[88] G. Welch and G. Bishop. An Introduction to the Kalman Filter. *University of North Carolina at Chapel Hill, Chapel Hill, NC*, 1995.

[89] Texas Instruments. http://www.ti.com/, Accessed Sept 2009.

[90] M. Lowe B. Pavlakoviec. *Disperse: User's manual.* Imperial College, Non-Destructive Testing Laboratory.

[91] C. Hellier. *Handbook of nondestructive evaluation.* McGraw-Hill Professional, 2001.

[92] Olympus NDT. http://www.olympus-ims.com/, Accessed Nov 2009.

[93] M. Castaings and P. Cawley. The generation, propagation, and detection of Lamb waves in plates using air-coupled ultrasonic transducers. *The Journal of the Acoustical Society of America*, 100:3070, 1996.

[94] J.L. Rose. *Ultrasonic Waves in Solid Media.* Cambridge University Press, 1999.

[95] Stephen P. Kelly. *An Air Coupled Ultrasonic Array Scannng System for Rapid Through Transmission NDT.* PhD thesis, University of Strathclyde, 2000.

[96] Roger Farlow. *Low Noise Techniques Applied to a Piezoceramic Receiver fo Gas Coupled Ultrasonic Flaw Detection.* PhD thesis, University of Strathclyde, 1998.

[97] CB Scruby and LE Drain. *Laser ultrasonics: techniques and applications.* Taylor & Francis, 1990.

[98] Maurice G. Silk. *Ultrasonic Transducers for Nondestructive Testing.* Adam Hilger Ltd, Bristol, 1984.

[99] SG Pierce and B. Culshaw. Laser generation of ultrasonic Lamb waves using low power opticalsources. *Science, Measurement and Technology, IEE Proceedings-*, 145(5):244–249, 1998.

[100] S. Dixon and SB Palmer. Wideband low frequency generation and detection of Lamb and Rayleigh waves using electromagnetic acoustic transducers (EMATs). *Ultrasonics*, 42(10):1129–1136, 2004.

[101] RB Thompson. Physical Principles of Measurements with EMAT Transducers. *Physical Acoustics*, 19:157–199.

[102] A. Gachagan. *An evaluation of 1-3 connectivity composite transducers for air-coupled ultrasonic applications.* PhD thesis, University of Strathclyde, 1996.

[103] Robert Banks. *A Frequency Agile Approach to Air-Couple lamb wave Inspection.* PhD thesis, University of Strathclyde, 1999.

[104] Gordon Hayward Stephen P. Kelly and Tomas E. Gomez Alvarez-Arenas. Characterization and assessment of an integrated matching layer for air-coupled ultrasonic applications. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 51:no. 10, 2004.

[105] B. Drinkwater and P. Cawley. The practical application of solid coupled ultrasonic transducers. *Materials Evaluation*, 55(3):401–406, 1997.

[106] Microchip. http://www.microchip.com/, Accessed Sept 2009.

[107] Microchip. *dsPIC DSC High-Performance 16-Bit Digital Signal Controllers*, rev 1 edition, 2005.

[108] EMCO. *Ultra-Miniature DC to HV DC Converters.*

[109] Analog Devices. *AD5930: Programmable Frequency Sweep and Output Burst Waveform Generator* , rev 0 edition, 2005.

[110] APEX. *PA19: VIDEO POWER OPERATIONAL AMPLIFIER*, rev h edition, 1998.

[111] Analog Devices. *AD811: High Performance Video Op Amp*, rev e edition, July 2004.

[112] 250mA High-Speed Buffer. Texas Instruments, September 2000.

[113] Analog Devices. *AD8058: Low Cost, Dual, High Performance Voltage Feedback, 325 MHz Amplifier* , rev b edition, August 2003.

[114] R. Mancini. *Op amps for everyone: design reference.* Newnes (an imprint of Butterworth-Heinemann Ltd), 2003.

[115] Texas Instruments. *Wideband, Ultra-Low Noise, Voltage-Feedback OPERATIONAL AMPLIFIER with Shutdown*, December 2008.

[116] ANSYS Inc. ANSYS. www.ansys.com, Accessed Dec 2008.

[117] The COMSOL Group. COMSOL. www.comsol.com, Accessed Dec 2008.

[118] Weildlinger Associates, Inc. PxFlex. www.pzflex.com, Accessed Dec 2008.

[119] PP Delsanto, T. Whitcombe, HH Chaskelis, and RB Mignogna. Connection machine simulation of ultrasonic wave propagation in materials. I: The one-dimensional case. *Wave motion*, 16(1):65–80, 1992.

[120] G. Hayward and J.A. Hossack. Unidimensional modeling of 1-3 composite transducers. *The Journal of the Acoustical Society of America*, 88:599, 1990.

[121] H. Zhang, DE Chimenti, and S. Zeroug. Transducer misalignment effects in beam reflection from elastic structures. *The Journal of the Acoustical Society of America*, 104:1982, 1998.

[122] S.D. Holland, S.V. Teles, and DE Chimenti. Air-coupled, focused ultrasonic dispersion spectrum reconstruction in plates. *The Journal of the Acoustical Society of America*, 115:2866, 2004.

[123] O.I. Lobkis and DE Chimenti. 3-D voltage model for detection of sound radiated from anisotropic materials. *Ultrasonics*, 38(1-8):237–241, 2000.

[124] B. Hosten and C. Biateau. Finite element simulation of the generation and detection by air-coupled transducers of guided waves in viscoelastic and anisotropic materials. *The Journal of the Acoustical Society of America*, 123:1963, 2008.

[125] B. Hosten and M. Castaings. Transfer matrix of multilayered absorbing and anisotropic media. Measurements and simulations of ultrasonic wave propagation through composite materials. *The Journal of the Acoustical Society of America*, 94:1488, 1993.

[126] M. Castaings, P. Cawley, R. Farlow, and G. Hayward. Single sided inspection of composite materials using air coupled ultrasound. *Journal of Nondestructive Evaluation*, 17(1):37–45, 1998.

[127] PP Delsanto, T. Whitcombe, HH Chaskelis, RB Mignogna, and R. Kline. Connection machine simulation of ultrasonic wave propagation in materials. II the two-dimensional case,”. *Wave Motion*, 20:295–314, 1994.

[128] PP Delsanto, RS Schechter, and RB Mignogna. Connection machine simulation of ultrasonic wave propagation in materials III: The three-dimensional case. *Wave Motion*, 26(4):329–339, 1997.

[129] BC Lee and WJ Staszewski. Modelling of Lamb waves for damage detection in metallic structures: Part I. Wave propagation. *Smart Materials and Structures*, 12(5):804–814, 2003.

[130] BC Lee and WJ Staszewski. Modelling of Lamb waves for damage detection in metallic structures: Part II. Wave interactions with damage. *Smart Materials and Structures*, 12(5):815–824, 2003.

[131] F. Moser, L.J. Jacobs, and J. Qu. Modeling elastic wave propagation in waveguides with the finite element method. *NDT and E International*, 32(4):225–34, 1999.

[132] E. Moulin, J. Assaad, C. Delebarre, and D. Osmont. Modeling of Lamb waves generated by integrated transducers in composite plates using a coupled finite element–normal modes expansion method. *The Journal of the Acoustical Society of America*, 107:87, 2000.

[133] DN Alleyne and P. Cawley. The interaction of Lamb waves with defects. *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, 39(3):381–397, 1992.

[134] J.C. Strikwerda. *Finite Difference Schemes and Partial Differential Equations.* Society for Industrial Mathematics, 2004.

[135] DN Alleyne and P. Cawley. A 2-dimensional Fourier transform method for the quantitativemeasurement of Lamb modes. *Ultrasonics Symposium, 1990. Proceedings., IEEE 1990*, pages 1143–1146, 1990.

[136] Roger Farlow Stephen P Kelly and Gordon Hayward. Applications of through-air ultrasound for rapid NDE scanning inthe aerospace industry. *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, 43(4):581–591, 1996.

[137] A. Gachagan, G. Hayward, SP Kelly, and W. Galbraith. Characterization of air-coupled transducers. *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, 43(4):678–689, 1996.

[138] G. Benny, G. Hayward, and R. Chapman. Beam profile measurements and simulations for ultrasonic transducers operating in air. *The Journal of the Acoustical Society of America*, 107:2089, 2000.

[139] James Hensman Stephen Gareth Pierce Rahul Summan, Gordon Dobie and Keith Worden. A probabilistic approach to robotic nde inspection. In *Proceedings of 2009 Review of Progress in Quantitative Nondestructive Evaluation*, Chicago, 2009.

[140] R. Farlow and G. Hayward. Real-time ultrasonic techniques suitable for implementing non-contact NDT systems employing piezoceramic composite transducers. *Insight*, 36(12):926–935, 1994.

[141] A.C. Kak and M. Slaney. *Principles of Computerized Tomographic Imaging.* IEEE Press New York, 1988.

[142] E.V. Malyarenko and M.K. Hinders. Fan beam and double crosshole Lamb wave tomography for mapping flaws in aging aircraft structures. *The Journal of the Acoustical Society of America*, 108:1631, 2000.

[143] P. Belanger and P. Cawley. Feasibility of low frequency straight-ray guided wave tomography. *NDT and E International*, 42(2):113–119, 2009.

[144] D. Hutchins. Lamb-wave tomography using non-contact transduction. *Ultrasonics*, 31(2):97–103, 1993.

[145] C. Holmes, B.W. Drinkwater, and P.D. Wilcox. Post-processing of the full matrix of ultrasonic transmit–receive array data for non-destructive evaluation. *NDT and E International*, 38(8):701–711, 2005.

[146] M. Karaman, P.C. Li, and M. O'Donnell. Synthetic aperture imaging for small scale systems. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 42(3):429–442, 1995.

[147] K. Nagai. A new synthetic-aperture focusing method for ultrasonic B-scan imaging by the Fourier transform. *IEEE Transactions on Sonics and Ultrasonics*, 32(4):531–536, 1985.

[148] D. Levesque, A. Blouin, C. Neron, and J.P. Monchalin. Performance of laser-ultrasonic F-SAFT imaging. *Ultrasonics*, 40(10):1057–1063, 2002.

[149] R. Sicard, J. Goyette, and D. Zellouf. A SAFT algorithm for lamb wave imaging of isotropic plate-like structures. *Ultrasonics*, 39(7):487–494, 2002.

[150] MM Bronstein, AM Bronstein, M. Zibulevsky, and H. Azhari. Reconstruction in diffraction ultrasound tomography using nonuniform FFT. *IEEE transactions on medical imaging*, 21(11):1395–1401, 2002.

[151] E.V. Malyarenko and M.K. Hinders. Ultrasonic Lamb wave diffraction tomography. *Ultrasonics*, 39(4):269–281, 2001.

[152] E.V. Malyarenko and M.K. Hinders. Fan beam and double crosshole Lamb wave tomography for mapping flaws in aging aircraft structures. *The Journal of the Acoustical Society of America*, 108:1631, 2000.

[153] Polytec. http://www.polytec.com/eur/default.asp, Accessed Dec 2008.

Appendix A

Circuit Diagrams

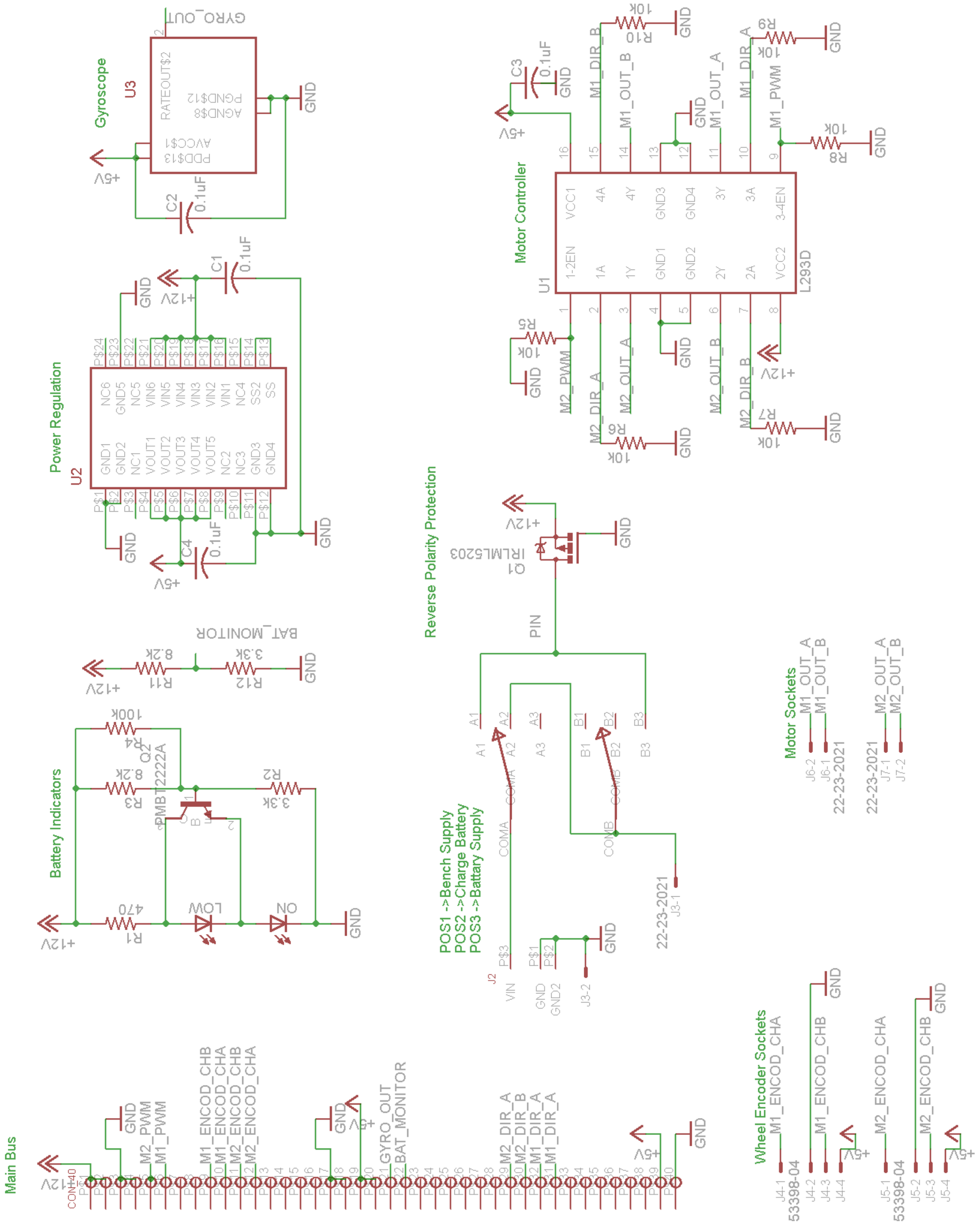


Figure A.1: Robot's motherboard schematic

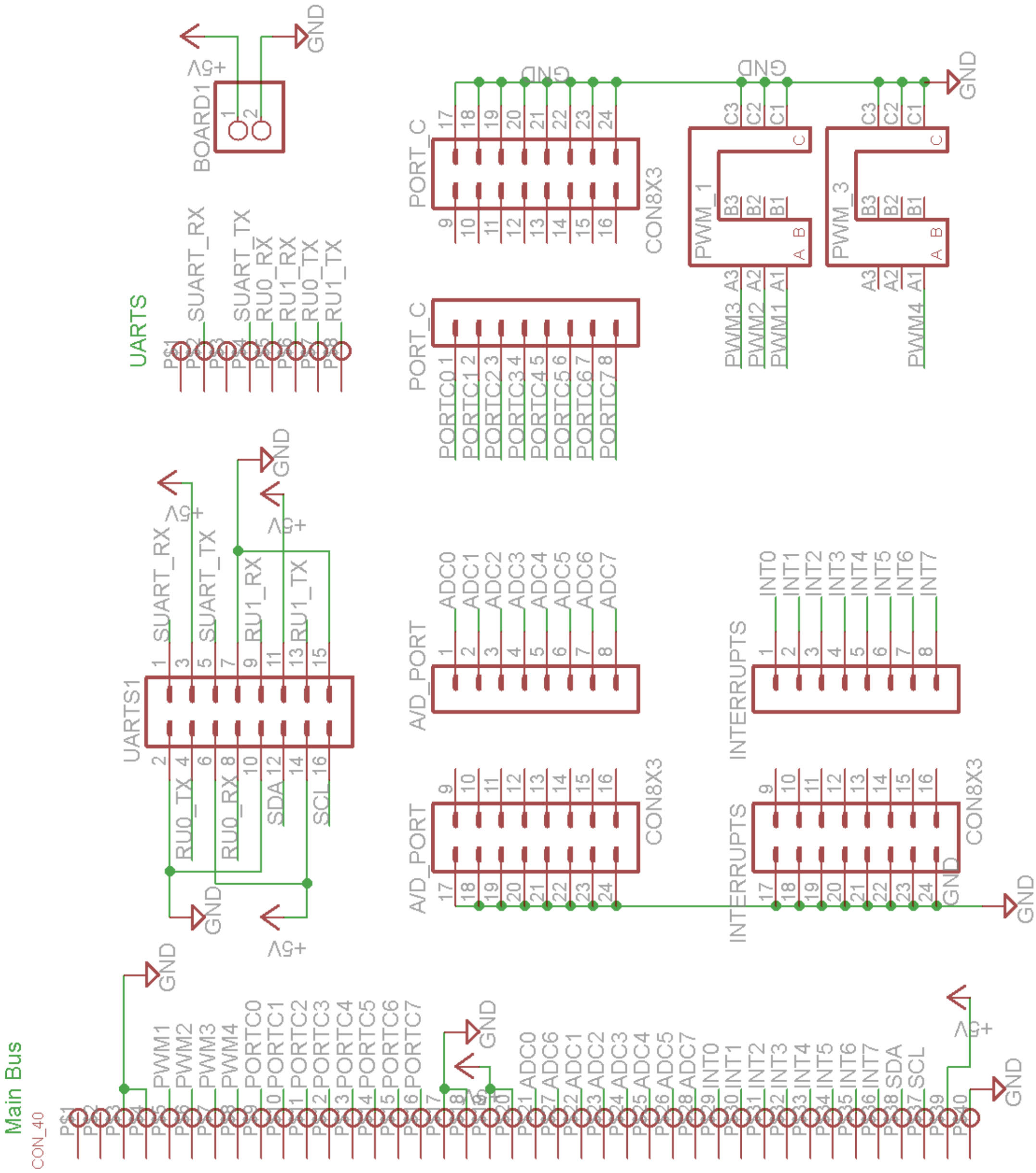


Figure A.2: Robots main processing board schematic

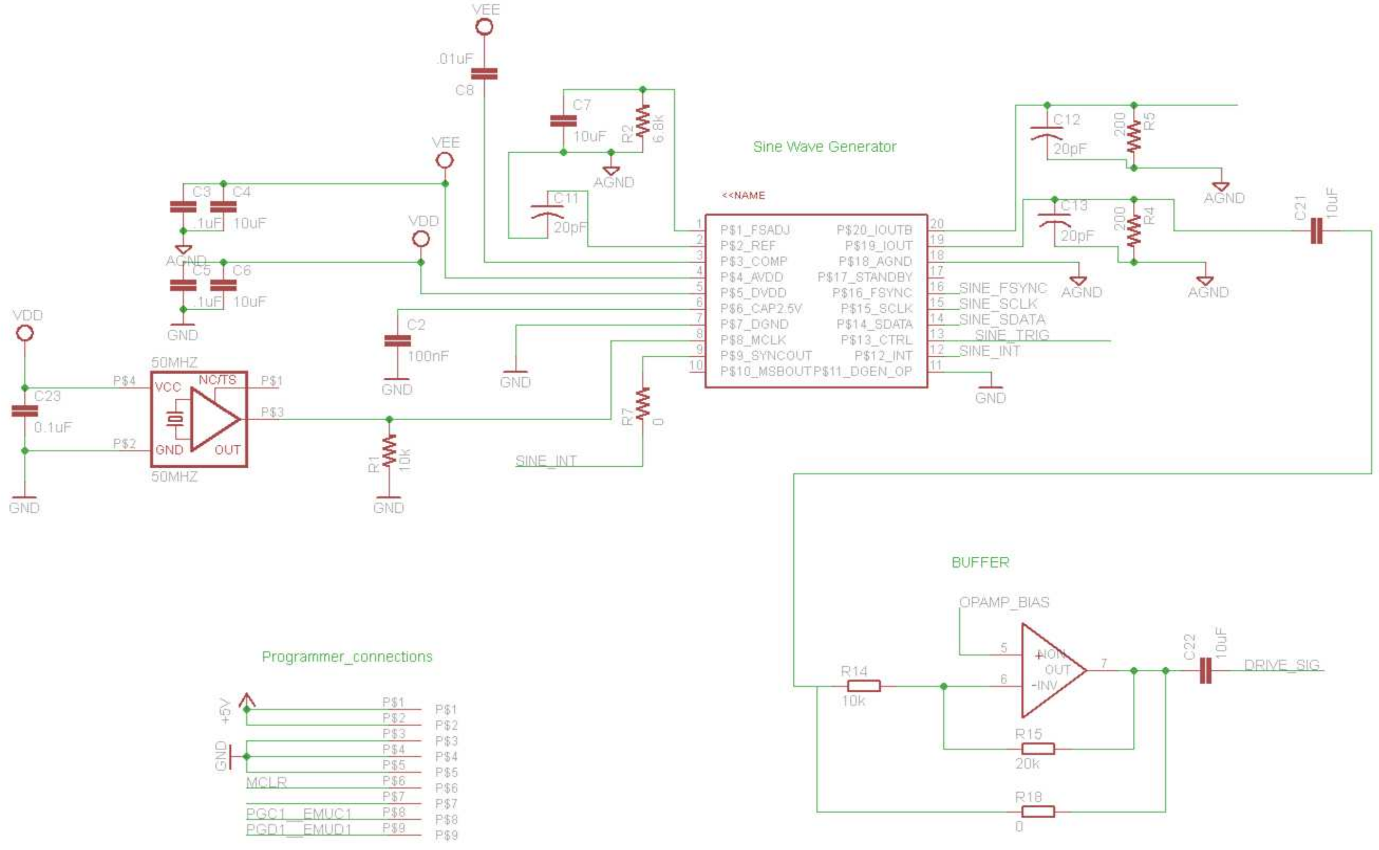


Figure A.3: Schematic of ultrasonic payload's digital circuitry - I of II

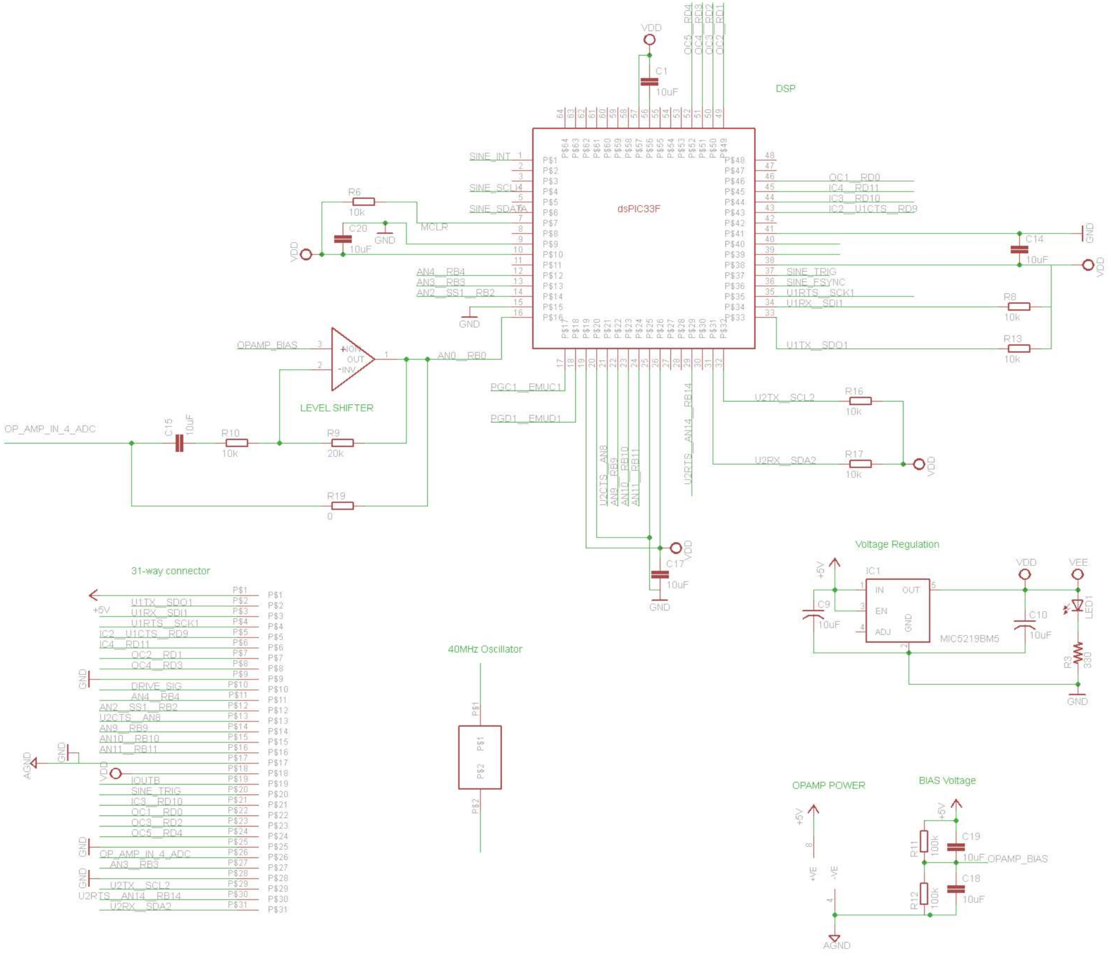


Figure A.4: Schematic of ultrasonic payload's digital circuitry - II of II

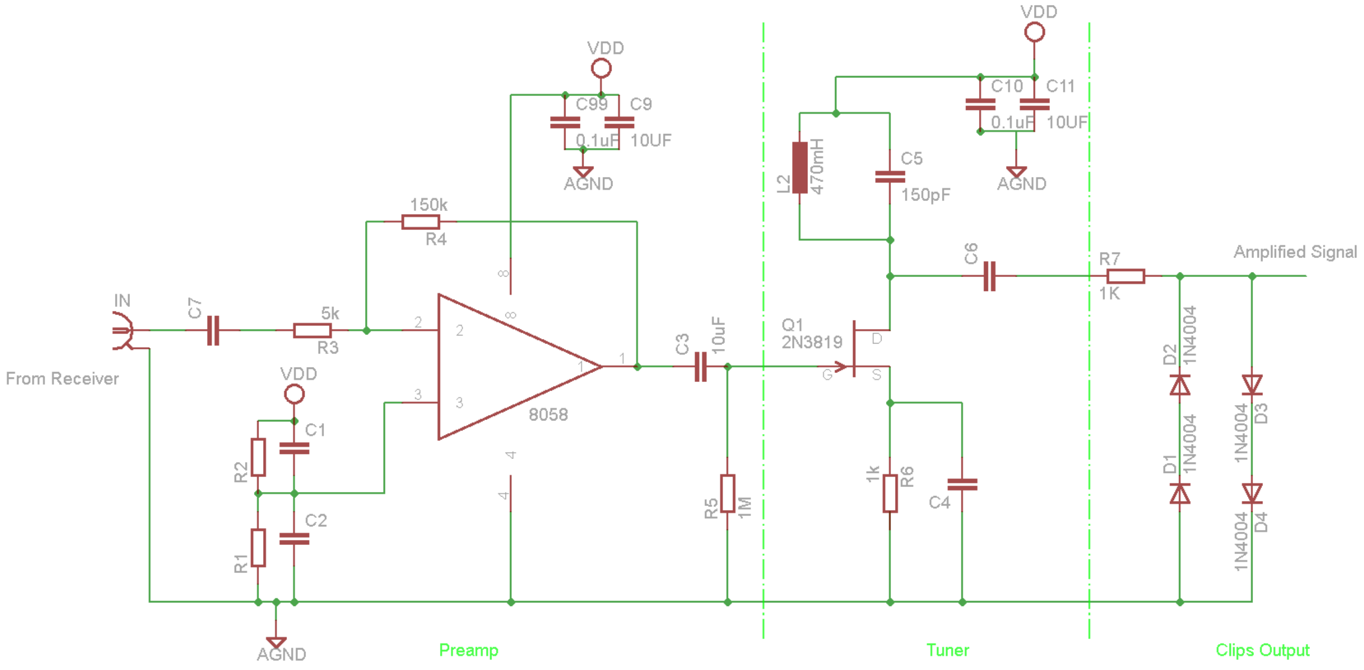


Figure A.5: Schematic of embedded preamplifier and tuner circuitry

Appendix B

Transformation between Cartesian Coordinate System

Rotational transforms are used extensively throughout this thesis to relate coordinate systems. One example is to transform coordinates from the payload's coordinate system to the global coordinate system. This is shown in Figure B.1.

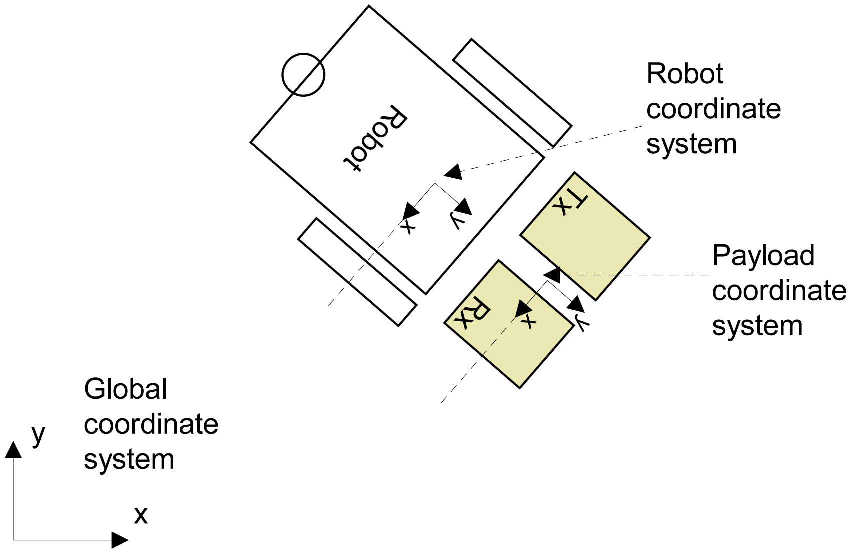


Figure B.1: Coordinate transformation example

The general transformation of a point or vector is shown in Equation B.1.

$$v' = R_\theta v; \tag{B.1}$$

where v' is the transformation of v and R_θ is transform. The following three transforms are used for a counterclockwise rotation by an angle θ relative to the x, y or

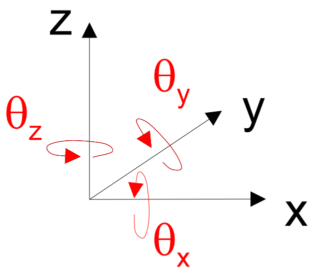


Figure B.2: Cartesian coordinates

z axis as defined in Figure B.2.
Rotation around X axis:

$$\mathbf{R_{CCW_x}}(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta & sin\theta \\ 0 & -sine\theta & cos\theta \end{bmatrix} \tag{B.2}$$

Rotation around Y axis:

$$\mathbf{R_{CCW_y}}(\theta) = \begin{bmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sine\theta & 0 & cos\theta \end{bmatrix} \tag{B.3}$$

Rotation around Z axis:

$$\mathbf{R_{CCW_z}}(\theta) = \begin{bmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{B.4}$$

To rotate in the clockwise direction replace θ with $-\theta$ and simplify noting that $cos(-\theta) = cos\theta$ and $sin(-\theta) = -sin\theta$.

Considering again the example in Figure B.1; to transform the measurement coordinates from the measurement coordinate system to the global coordinate system the following steps are required:

1. Transform from the measurement coordinate to the robot coordinate system. In this case it is a simple translation.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}' = \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix}_{measurement}$$

2. Apply a clockwise rotation around the ‘z’ axis to account for the robot’s angular offset from the global coordinate system.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}'' = \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y_o \\ z \end{bmatrix}'$$

3. Apply a translation to account for the offset between the robot and global coordinate system.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}''' = \begin{bmatrix} x \\ y \\ z \end{bmatrix}'' + \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix}_{robot}$$

x”’ = **x** transformed to the global coordinate system.