Development of a Human Accompanying Wheelchair using Ultrasonic Tethering

Theja Ram Pingali

Thesis submitted to the Faculty of Engineering
in partial fulfillment of requirements for the degree of

Master of Applied Science

in

Biomedical Engineering

Ottawa Carleton Institute for Biomedical Engineering

University of Ottawa

Ottawa, Ontario

Abstract

In social situations, people who use a powered wheelchair must divide their attention between navigating the chair and conversing with people. As a solution that maintains a good conversation distance between the wheelchair and the accompanying person, a wheelchair control system was introduced to provide automated side-by-side following by wirelessly tethering the wheelchair to the person.

This thesis designed, developed, and evaluated a wireless tethering system using ultrasonic sensors. Two ping sensors and three piezoelectric ultrasonic transducers were used to identify the accompanying person and determine their pose. A trajectory algorithm determined the person's direction of motion and a drive control algorithm determined the wheelchair's required direction by maintaining a comfortable conversation distance between the person and the wheelchair user. A plug-and-play prototype was developed using commercially available components and the firmware was implemented using an open-source platform. The prototype developed in this thesis was mounted to a Permobil F3 Corpus powered wheelchair with a modified Eightfold Technologies SmartChair Remote, which controlled the wheelchair direction.

Results demonstrated that the system can navigate a wheelchair beside an accompanying person and maintain a comfortable conversation distance, which is advantageous for users who require hands-free wheelchair control during social activities.

Table of Contents

Abstract ..... **ii**

Table of Contents ..... **iii**

List of Figures ..... **vii**

List of Tables ..... **xi**

Abbreviations and Definitions ..... **xii**

Acknowledgments ..... **xiv**

1 Introduction **1**

1.1 Rationale 1

1.2 Objectives 2

1.3 Thesis contributions 2

1.4 Thesis outline 2

2 Literature Review **4**

2.1 Overview 4

2.2 Powered wheelchair 4

2.3 Operating a powered wheelchair 4

2.4 Wheelchair-related accidents 5

2.4.1 Tips and falls 5

2.4.2 Accidental contact 6

2.4.3 Dangerous operations 6

2.5 Smart wheelchairs 6

2.5.1 Form factor 7

2.5.2 Control system 7

2.5.3 Operating modes 9

2.5.4 Obstacle detection/avoidance and assistive navigation 9

2.5.5 Sensors for obstacle avoidance 10

2.5.6 Mapping, localization, and path planning 13

2.5.7 Path planning algorithms 15
2.5.8 Human following (tethering) 16
2.5.9 Human following in mobile robots 25
2.6 Summary 27
3 Modelling and Simulation 29
3.1 Overview 29
3.2 Abstract 29
3.3 Introduction 30
3.4 Model design and development 30
3.4.1 Accompanying person sub-system 32
3.4.2 Ultrasonic sensor module 34
3.4.3 Piezoelectric ultrasonic receiver module 36
3.4.4 Processing unit 36
3.5 Results 37
3.6 Conclusions 40
4 Ultrasonic Tethering to Enable Side-by-Side Following for Powered Wheelchairs 41
4.1 Overview 41
4.2 Abstract 41
4.3 Introduction 41
4.4 Methodology 43
4.4.1 System architecture 44
4.4.2 Identifying the accompanying person 44
4.4.3 Determining accompanying person position and orientation 45
4.4.4 Identification and pose detection algorithm 47
4.5 Prototype 49
4.5.1 Accompanying person beacon 50
4.5.2 Receiver on the powered wheelchair 50
4.5.3 Effect of accompanying person's height on the tethering system 51
4.5.4 Accompanying person feedback and ultrasonic tethering system integration with the powered wheelchair 52

4.5.5 Sensor calibration 53
4.6 Ultrasonic tethering system experimental test protocol 53
4.7 Results 54
4.8 Discussion 56
4.9 Conclusion 57
5 Smart Wheelchair Navigation using Ultrasonic Tethering for Social Following 58
5.1 Overview 58
5.2 Abstract 58
5.3 Introduction 59
5.4 Identifying and determining the accompanying person's pose 61
5.5 Sensor crosstalk and its effects on the identification algorithm 62
5.5.1 Identification algorithm I 62
5.5.2 Identification algorithm II 63
5.6 Wheelchair drive control concept 63
5.6.1 Wheelchair trajectory and speed algorithm 64
5.6.2 Wheelchair navigation control algorithm and the SmartChair Remote 66
5.6.3 SmartChair Remote 66
5.6.4 User selected powered wheelchair speed settings 67
5.6.5 Wheelchair navigation – direction and speed commands 68
5.6.6 Hard and soft turns using the SmartChair Remote 69
5.7 Ultrasonic tethering system testing 70
5.7.1 Straight path test 70
5.7.2 Mixed path test 70
5.8 Results 72
5.8.1 Calibration results 73
5.8.2 Straight path test results 73
5.8.3 Mixed path test results 76

5.9 Discussion 79

5.10 Conclusions 82

6 Thesis Conclusions and Future Work 83

Objective 1: 83

Objective 2: 83

6.1 Future work 83

6.1.1 Improved trajectory and speed algorithm 84

6.1.2 Drive control for backward navigation and sharp left and right turns 84

6.1.3 Self-localization using proprioceptive sensors 84

6.1.4 Smartphone beacon 85

6.1.5 Cross compatibility to other joystick emulators 85

6.1.6 Error detection and fail-safes 85

References 86

Appendix A 93

1 Bill of materials: Beacon circuit (components, descriptions and parameters) 97

2 Bill of materials: Receiver circuit (components, descriptions and parameters) 97

Appendix B 99

Appendix C 108

1 Ultrasonic tethering system hardware block diagram 108

2 Variables and definitions 109

3 Rule based algorithm to calculate wheelchair direction 110

4 Beacon software flow diagrams 111

5 Receiver software flow diagram 112

List of Figures

Figure 2.1 (a) VAHM [20], (b) NavChair [21], and (c) SmartChair controller [23]. 7

Figure 2.2 Typical smart wheelchair control system [34]. 9

Figure 2.3 Obstacle avoidance system with mobile robot avoiding walls by detecting a solid structure in its global path towards the goal. 10

Figure 2.4 (a) Photoelectric and ultrasonic sensors mounted under the wheelchair carriage facing outward (360-degree field of view) [22], (b) Modulated time of flight [37], and (c) Pulsed time of flight [37]. 10

Figure 2.5 Door-crossing detection algorithm: (a) Door-way is left of wheelchair, (b) Door-way center of wheelchair, and (c) Door-way right of wheelchair [42]. 14

Figure 2.6 Generalized path planner algorithm. The green path consists of straight lines and sharp turns at A, B, C, D, and E. The green path is generated by the path planning algorithm. The smooth path generated by the path smoothing algorithm is in red [41]. 15

Figure 2.7 Wired/contact human tethering [43]. 17

Figure 2.8 (a) Wired/contact human tethering between a human and robot with tether length, orientation angles, and positions, and (b) Prototype for tether tension and angle [42]. 18

Figure 2.9 (a) Robotic wheelchair prototype for moving alongside a caregiver, (b) Omni-directional camera and an LRS attached to the wheelchair's back [47]. 19

Figure 2.10 (a) Caregiver shoulder contour modeled as coordinate points in 2D space, and (b) caregiver tracking using the maximum distance between shoulder contours [47]**.** 20

Figure 2.11 (a) JiaoLong wheelchair with driver assistance and human following [48], and (b) Motion vector generation based on the direction of obstacles and person to follow [47]. 22

Figure 2.12 Guide following smart wheelchair prototype using a laser range sensor [48]. 23

Figure 2.13 (a) Grid map generated by the control system (LRS field of view as gray cells in a coordinate plane), and (b) Waypoint and path generation based on guide position [48]. 24

Figure 2.14 (a) Human following mobile robot based on LRS, (b) Human following mobile robot with web camera, LRS, and laptop computer [44], and (c) LRS scanning of the person's shin for different person orientation angles [43]. 26

Figure 3.1 (a) Sensor placement on the wheelchair and beacon [53], and (b) Ultrasonic tethering model in Simulink. Ping sensor transmits and receives ultrasonic pulses. Piezoelectric receiver receives ultrasonic pulses. 31

Figure 3.2 Accompanying person sub-system for model identification, walking, turning, and stopping. ..32

Figure 3.3 Ultrasonic signals produced by the front, back and center generic linear sensors. Signals are amplitude modulated square wave pulses of 42 KHz with 50% duty cycle. 33

Figure 3.4 The 40 KHz 10 V p-p pulse generator module that consists of a voltage doubler, a voltage inverter, an oscillator and a PWM generator. 33

Figure 3.5 The ping sensor module for determining pose that consists of a multiplexer, two Simulink generic linear sensors (modelled as ping sensors). 34

Figure 3.6 (a) Noisy signal generated by front and back ping sensors and, (b) Ultrasonic sensor amplification and filtering circuit that consists of an instrumentation amplifier and low-pass filter circuits. 35

Figure 3.7 (a) Piezoelectric ultrasonic receiver module in Simulink, and (b) noizy signal produced by the piezoelectric ultrasonic receiver. 36

Figure 3.8 The Matlab function block that simulated a processing unit....37

Figure 3.9 Sample output from the Parameter Estimation tool to optimize the gain resistor Rg used for different conditions. The parameters converge at around 100 Ohm at the seventh iteration. 38

Figure 3.10 (a) Post-processed front (orange) and back (magenta) ping sensor signals to calculate tether distance and calculated tether distance (blue), and (b) Direction of left and right wheelchair motors. 39

Figure 4.1 (a) Ultrasonic tethering system prototype mounted on a Per4mobil F3 corpus powered wheelchair, (b) and (c) Ultrasonic tethering system for social following. 44

Figure 4.2 Detection area beside the powered wheelchair for AP identification. Beam patterns converge at the center of where the person is expected to walk. 45

Figure 4.3 AP position in coordinate space beside the wheelchair. x and y represent location coordinates and θ is the angle of a line from the origin to the point x, y. 46

Figure 4.4 Relationship between the AP conversation distance (d_c) and tether distance (D). T and L are distances between the Ping sensors and AP. α and β are angles between Ping sensors and AP. M is distance between Ping sensors. d1 is distance from wheelchair sagittal plane to the receiver module. d2 is distance from AP sagittal plane to beacon. 46

Figure 4.5 Ultrasonic transducer analog output while the person is (a) walking and stopping, (b) turning right, (c) turning left. Dashed line is high threshold.....47

Figure 4.6 (a) Signals from front and back receivers for three trials. (b) Thresholding used on the center receiver signal to determine if the person is in-motion or static for three trials. Dashed lines represent the thresholds. 48

Figure 4.7 Nomenclature used for the algorithm. M is the distance between the front and back sensors. T and L are distances from the sensors to the AP side. α and β are angles formed between M and T or L and θ is the angle formed between M and the tether distance.....48

Figure 4.8 (a) Ultrasonic beacon for AP, (b) ultrasonic tethering receiver for powered wheelchair (not to scale).....49

Figure 4.9 Receiver module hardware block diagram.....50

Figure 4.10 (a) The piezoelectric transducer mean and standard deviation signal output for beacon heights ranging between 80 cm to 98 cm, (b) the ultrasonic tethering receiver mounted to the wheelchair using a movable hinge, and (c) the beacon with a movable hinge.....51

Figure 4.11 Ultrasonic tethering system performance test setup.....54

Figure 4.12 (a) Measured tether distance for the three walk trials. The dashed line represents the experimental tether distance (30 cm). The brackets indicate the effect of AP walking on the calculated tether distance, observed as sinusoidal waves, and (b) Wheelchair direction commands for the three walk trials. 55

Figure 5.1 The ultrasonic tethering receiver mounted on a Permobil F3 Corpus wheelchair. 60

Figure 5.2 Comparison between distances from the receiver with the beacon on and off. 62

Figure 5.3 Receiver and beacon on, and off time intervals (states).....63

Figure 5.4 Representation of an overall path between the starting position (A) and the destination (E), with multiple smaller paths of fixed directions (A to B; B to C; etc.).....64

Figure 5.5 (a) The SmartChair Remote, (b) the SmartChair Remote smartphone app that can be used to control wheelchair direction, and (c) SmartChair Remote mounted to a powered wheelchair [23]. 67

Figure 5.6 (a) Speed 1 (low speed) setting and, (b) Speed 5 (high speed) setting and the wheelchair speed selector interface on the manual joystick controller.....68

Figure 5.7 Wheelchair directions by moving the joystick to the (a) hard right (east), (b) soft right (north-east), (c) hard left (west), and (d) soft left (north-west). 69

Figure 5.8 Ultrasonic tethering system, the SmartChair Remote and the joystick controller. 70

Figure 5.9 (a) The straight path test setup, (b) mixed path A and B, (c) mixed path C and D. 71

Figure 5.11 Processed video screenshots from the straight path test collected from a fixed camera (main video) and the smartphone camera mounted to the wheelchair (top right video overlay). 73

Figure 5.12 Straight path test results for 10 trials. (a) AP tether distance during walking (purple), Precomputed tether distance (dashed red), threshold band (low and high threshold calculated during calibration, red band) and, (b) tether distance corresponding wheelchair direction commands (f – Forward, l – Left, r – Right, s – Stop) produced by the UT system for each trial. 74

Figure 5.13 Straight path test results for 10 trials. (a) AP orientation during walking (purple), Precomputed tether angle (dashed red), threshold band (low and high thresholds calculated during calibration, red band), and (b) wheelchair speed (0 – 70% of the user selected speed). 75

Figure 5.14 Processed video screenshots from the mixed path test collected from a fixed camera (main video) and the smartphone camera mounted to the wheelchair (bottom right video overlay). 76

Figure 5.15 Mixed path test results. (a) AP tether distance during walking (purple), Precomputed tether distance (dashed red), threshold band (low and high threshold calculated during calibration, red band), (b) wheelchair direction commands (f – Forward, l – Left, r – Right, s – Stop) produced by the UT system. The four paths are labeled as A to D beside each graph. 77

Figure 5.16 Mixed path test results. (a) AP orientation during walking (purple), Precomputed tether angle (dashed red), threshold band (low and high thresholds calculated during calibration, red band) and, (b) wheelchair speed (0 – 70% of the user selected speed). The four paths are labeled as A to D beside each graph. 78

Figure 5.17 (a) AP turning left along BD and the wheelchair following the AP along AC, (b) AP turning right along BD and the wheelchair following the AP along AP. 80

Figure 5.18 Screenshots of the straight walk test during unidentification, where (a) the AP walked faster than the UT system's window thus causing an identification error, (b) the AP had to take a step back to be re-identified. 81

List of Tables

Table 2-1 Input-control modes implemented in smart wheelchairs.....8

Table 2-2 Advantages and disadvantages of time-of-flight range sensors. 11

Table 2-3 Human following technologies implemented in wheelchairs. 28

Table 3-1: Optimized component parameters using the Parameter Estimation tool and the final chosen values for each component. 38

Table 3-2: Optimized thresholds calculated from the average tether distance for each AP direction. 39

Table 4-1 Ultrasonic tethering system modes of operation. 52

Table 4-2 The low and high thresholds and the wheelchair direction command errors produced by the drive control for each trial. 56

Table 5-1 (a) UT system direction commands with the wheelchair direction, and (b) Speed commands with the wheelchair speed (in % of the maximum user selected speed on the joystick). 68

Table 5-2 Straight path test results: the tether distance threshold (T-D) and tether angle (T-A), wheelchair direction commands (F – forward, L – left, R – right, S – stop) in percent of all commands transmitted to the SmartChair Remote (using equation 5.7). 76

Table 5-3 Mixed path test results: The AP paths used for each trial, the precomputed tether distance (T-D) and tether angle (T-A) used for calculating the wheelchair's trajectory, and the number of times the system produced an identification error. 79

Abbreviations and Definitions

2-D	Second Dimension
3-D	Third Dimension
AD	Alzheimer Disease
ADC	Analog to Digital Converter
ALS	Amyotrophic Lateral Sclerosis
AP	Accompanying Person
ARM	Advance RISC Machine
BJT	Bipolar Junction Transistor
CAD	Computer Aided Designs
CMBEC	Canadian Medical and Biological Engineering Society
DIP	Dual In-line Package
DSP	Digital Signal Processing
DOF	Degree of Freedom
EKF	Extended Kalman Filter
EMG	Electromyography
FoV	Field-of-View
GPS	Global Positioning Systems
IMU	Inertial Measurement Units
LRS	Laser Range Sensor/Scanner
MS	Multiple Sclerosis
MVFH	Minimum Vector Field Histogram
PCB	Perforated Circuit Board/Printed Circuit Board
PI	Proportional and Integral
PID	Proportional, Integral and Differential
PLA	Polylactic Acid
PW	Powered Wheelchair
RIWS	Robotic Intelligent Wheelchair System
RLC	Resistor, Inductor and Capacitor
ROS	Robot Operating System

RS-232	Recommended Standard 232
VFF	Vector Force Field
SIP	Single In-line Package
SLAM	Simultaneous Localization and Mapping
SPDT	Single Pole Dual Throw
SW	Smart Wheelchair
SWCS	Smart Wheelchair Component System
TBI	Traumatic Brain Injury
TH	Through Hole
ToF	Time of Flight
TO	Transistor Outline
TTL	Transistor-Transistor Logic
UT	Ultrasonic Tethering
VAHM	Véhicule Autonome Pour Handicapé Moteur

Acknowledgments

I would first like to express my sincere gratitude to my supervisors, Dr. Edward D. Lemaire and Dr. Natalie Baddour for giving me an opportunity to work on this thesis. I thank them for guiding me throughout the thesis, especially during trouble spots. Their invaluable support and encouragement are immeasurable, and I will be forever grateful.

I would like to acknowledge Tony Awad and Ke Wang from Eightfold Technologies for lending us a SmartChair Remote and their guidance in using the same, and the Natural Sciences and Engineering Research Council of Canada for funding the research of this thesis.

I would like to express my profound gratitude to my parents, Phani Ram Pingali and Padmini Pingali, and my brothers, Govinda Ram Pingali and Dr. Karthik Ram Pingali for their unconditional love, patience and support throughout my life. My accomplishments would not have been possible without them.

Finally, I would like to thank all my friends and the staff at The Ottawa Hospital Rehabilitation Centre for all the encouragement through the process of researching and writing this thesis. Thank you.

1 Introduction

Powered wheelchairs were developed to benefit people with mobility disorders, reducing the strain of using a manual wheelchair and improving productivity and quality of life by increasing leisure and social participation [1]–[4]. Traditional joystick controlled powered wheelchair navigation is a complex and dynamic task that requires adequate motor skills for controlling and navigating, and good cognitive reasoning to interpret and act upon the driving environment. In social situations where a person accompanies a powered wheelchair user, conversations between the wheelchair user and the accompanying person could result in distracted navigation, potentially leading to accidents.

In an effort to increase powered wheelchair user's safety and comfort, researchers have implemented mobile robot navigation for powered wheelchairs, thus creating smart wheelchairs. Users with varying mobility needs require special types of smart wheelchair controls; such as, tongue, eye tracking using camera, semi-automatic and automatic navigation, and human-following [5]. Researchers have been pressed to develop human-following smart wheelchairs with hands-free navigation when moving behind a caregiver/guide or moving beside the person and participating in social interactions, thus reducing distractions while navigating.

The goal of this project was to improve wheelchair user safety by developing a smart wheelchair add-on unit that enables side-by-side human-following for users that may require navigation assistance in social situations.

1.1 Rationale

Current human-following technologies implemented in smart wheelchairs are only capable of following behind a person. This does not allow wheelchair users to appropriately converse with the accompanying person, which could lead to social anxiety. Furthermore, current human-following technologies use complex and bulky sensors and controllers that require extensive modifications to the powered wheelchair and cannot be easily transferred between different powered wheelchairs. Human following can use photoelectric sensors such as Lidar and laser range scanners (LRS) that require high processing power for computation, greatly increasing system cost. Wheelchair users may have difficulty accessing these systems due to cost constraints. Considering these views, a smart navigation system should follow an accompanying person from the side to

enable conversations between the wheelchair user and person, be cost effective, and be easy to mount on any commercially available powered wheelchair.

Ultrasonic tethering can be considered for human-following and wheelchair navigation. This technique provides contactless tethered steering between a powered wheelchair and an accompanying person (AP) using robust ultrasonic sensors to detect and calculate AP location for side-by-side following. Ultrasonic sensors are cost effective, have a small form factor, and are reliable for short distances.

1.2 Objectives

This thesis developed and evaluated a human-following wheelchair system using ultrasonic sensors (ultrasonic tethering). The objectives were:

1. Develop an ultrasonic tethering prototype to enable side-by-side following in social situations.
2. Test and validate the ultrasonic tethering prototype for smooth hands-free powered wheelchair navigation.

1.3 Thesis contributions

The contributions of this thesis were developing and testing the viability of a side-by-side human-following system for powered wheelchairs in social situations. This thesis provides a solution for human following that can be plugged into any commercially available powered wheelchair using an Eightfold Technologies SmartChair Remote.

1.4 Thesis outline

This thesis is divided into six chapters and two Appendices. Chapter 2 is a literature review that discusses the prevalence and demographics of powered wheelchair use, types of accidents that may occur when using powered wheelchairs, the need for smart wheelchairs and its types, and current human-following smart wheelchair technologies.

Chapter 3 presents and discusses the modeling and simulation of a human-following smart wheelchair system using ultrasonic tethering. This chapter contains a manuscript accepted by the Canadian Medical and Biological Engineering Society (CMBEC), Ottawa, Ontario, Canada (May 2019).

Chapter 4 contains a modified journal manuscript published in *Sensors* (2019, 19(1), p. 109) that presents the design and development of an ultrasonic tethering human-following smart wheelchair prototype. This chapter discusses ultrasonic tethering design considerations, prototype development and presents initial testing on a Permobil F3 Corpus powered wheelchair.

Chapter 5 discusses the powered wheelchair path planning algorithm and trajectory control implemented in the human-following ultrasonic tethering system discussed in Chapters 3 and 4.

Chapter 6 presents a thesis summary, thesis contributions, and suggestions for future work.

Appendix A contains the ultrasonic tethering circuit schematics. Appendix B contains 2D computer aided designs (CAD) of the 3D schematics and images for the prototype designed during the thesis. Appendix C contains hardware block diagrams and flow charts that represent the ultrasonic tethering system software.

2 Literature Review

2.1 Overview

This chapter reviews smart wheelchair requirements based on the wheelchair demographics and issues caused by dependency on a powered wheelchair. An overview of current smart wheelchair technologies for assistive navigation is provided to better determine requirements and gain insight for social following using ultrasonic tethering.

2.2 Powered wheelchair

Ninety percent of all assistive mobility users propel a manual wheelchair, which can cause repetitive strain injuries to the upper-body and lead to a sedentary lifestyle due to difficulties with maneuvering a manual wheelchair [1], [6]. Powered wheelchairs were developed to benefit people from mobility disorders, reducing the strain of using a manual wheelchair and improving productivity and quality of life by increasing leisure and social participation [1]–[4].

The prevalence of powered wheelchair users has steadily increased due to the ageing population and increases in chronic health conditions [1]. In 2004, Statistics Canada reported that 0.3% of people aged 12 to 44 years used a wheelchair (manual and powered), and the percentage of wheelchair users aged 85 years or older was 7% of the aged population [7], [8]. In 2012, 0.2% of the overall population used a powered wheelchair for mobility [9], [10]. Without a powered wheelchair, many people would be dependent on others to complete life tasks and would be unable to have independent mobility. Therefore, it is reasonable to state that the demand for powered wheelchairs is increasing [11].

2.3 Operating a powered wheelchair

Powered wheelchair operation requires adequate motor skills for controlling and navigating, and good cognitive reasoning to interpret and act upon the driving situation. Older adults, especially those in institutional settings, may have cognitive impairments that affect their ability to drive a powered wheelchair properly and safely. Cognitive impairments such as Alzheimer's disease and severe traumatic brain injury (TBI) are known to cause impaired attention, agitation, and poor impulse control [12], [13] that could affect executive reasoning.

For example, this can manifest as being unable to interpreting the environment during navigation and having difficulty concentrating on wheelchair navigation. Additionally, the complexity of powered wheelchair navigation increases when dividing attention between navigation and a second task. These problems increase as the complexity of powered wheelchair navigation increases [14].

Motor skills could be affected by diseases such as Alzheimer's disease, Amyotrophic Lateral Sclerosis (ALS), and Multiple Sclerosis (MS). Muscle weakness can interfere with the person's ability to operate a powered wheelchair [14], [15], leading to dangerous operation due to fatigue in the arms, shoulders, or neck [16]. The increase in complexity of powered wheelchair navigation or dangerous powered wheelchair operation could increase the risk of accidents.

2.4 Wheelchair-related accidents

Wheelchair accidents can be defined as "an event that interrupts normal wheelchair operation and either causes injury or poses the threat of injury in the user's judgment" [17]. These events may affect user mobility and can result in restrictions in activity and social participation caused by non-fatal or fatal user injuries [16]. A study with 95 manual and powered wheelchair users reported 74 wheelchair accidents within the past three years, of which 24 (32.4%) were powered wheelchair accidents [16]. According to a 2001 survey by the U.S. Consumer Product Safety Commission, 24.6% of the total 85,263 wheelchair accidents involved a powered wheelchair [18]. These accidents can be categorized as tips and falls, accidental contact, and dangerous operations [16] [19].

2.4.1 Tips and falls

Tipping is an event where one or more wheels of the wheelchair leave the ground, which may cause the user to fall off the wheelchair. Tipping and falling could directly or indirectly injure the user and the wheelchair. A 1994 survey on wheelchair accidents reported a wheelchair user falling out of his wheelchair while cooking, resulting in a fire and the person suffocating to death due to the smoke. 20%-30% of all wheelchair falls result in user injury requiring medical attention [18]. Fatal and non-fatal injuries (hip fractures, etc.) have a major impact on the well-being of older adults using powered wheelchairs.

2.4.2 Accidental contact

Accidental contact is an event where the wheelchair collides with a movable or immovable object (i.e., signposts, vehicles, people, etc.) but may not result in tips and falls. Accidental contact could injure the user, the powered wheelchair, or contacted object. For example, in 1994 a powered wheelchair user accidentally ran over a 10-month old baby, causing facial injuries [18]. Powered wheelchair damage during accidental contact affects quality of life due to repair difficulties, wheelchair unavailability during the repair period, and repair costs.

2.4.3 Dangerous operations

Dangerous operations are defined as an event where the user is unable to properly operate the wheelchair and could lead to quick changes in powered wheelchair direction or uncontrolled powered wheelchair navigation. For example, a user was electrocuted when he drove over a wet floor with the powered wheelchair still attached to the battery charger [18].

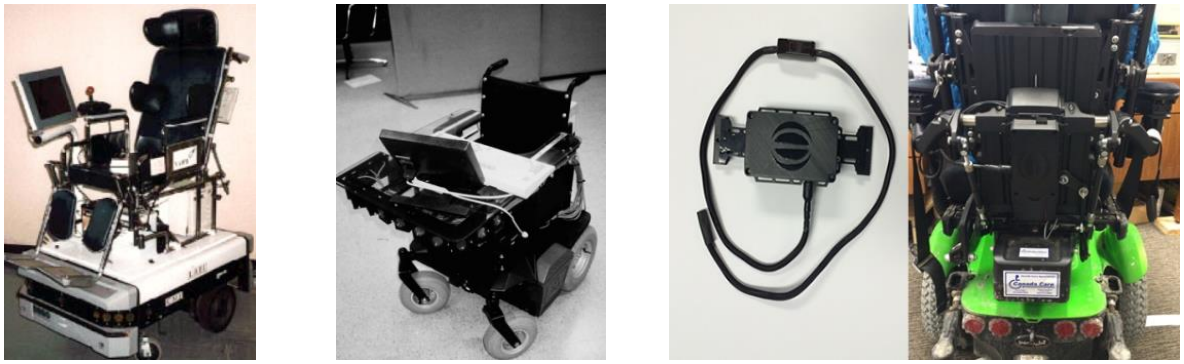
In a survey reporting prevalence, mechanisms, and wheelchair-using behaviors associated with wheelchair-related accidents, 87.8% of all wheelchair accidents were tips and falls, of which 2% occurred while using a powered wheelchair and 33% of the powered wheelchair accidents were accidental contact and dangerous operations [16]. Although the total number of participants for this survey were limited to 95, other research indicated that the prevalence of powered wheelchair accidents is increasing and can cause powered wheelchair damage and fatal or non-fatal injuries to users [18].

2.5 Smart wheelchairs

Recent technology advances have enabled researchers to implement mobile robot navigation and controls to powered wheelchairs, which gave rise to smart wheelchairs. Smart wheelchairs use autonomous navigation, anti-collision, assistive navigation, and path planning to improve powered wheelchair accessibility by increasing overall functionality [5]. To address safety concerns, smart wheelchairs are designed to enable independent mobility for users suffering from cognitive and motor impairments. Typically, smart wheelchairs consist of traditional or non-traditional input controls (e.g., joystick or gesture-based control), environmental and location sensors, a processor (control board), and a powered wheelchair. Smart wheelchairs can be classified based on their physical appearance, technology, and functionality [5].

2.5.1 Form factor

Initially, smart wheelchairs used a robotic base with a wheelchair seat attached. This allowed researchers to develop wheelchair assistive navigation systems with readily available mobile robot technology; for example, the Véhicule Autonome pour Handicapé Moteur (VAHM) (Figure 2.1) [20]. As technology progressed, the majority of smart wheelchairs were developed as complete units, consisting of a control unit and sensing systems connected to modified commercially available powered wheelchairs [5]. The NavChair (Figure 2.1) [21] was developed on a commercially available Lancer powered wheelchair to assist the user in navigation. Another approach connects an add-on unit to a standard powered wheelchair, to allow users to switch between powered and manual control. The Smart Wheelchair Component System (SWCS) [22] is a modular smart system with assistive navigation and the Eightfold Technologies SmartChair is a Bluetooth enabled remote controller that connects to most powered wheelchairs (Figure 2.1).



(a) **(b)** **(c)**

Figure 2.1 (a) VAHM [20], (b) NavChair [21], and (c) SmartChair controller [23].

2.5.2 Control system

Control systems consists of wheelchair input, sensing system, and processing system [5] [23]. Traditional powered wheelchairs use a joystick to maneuver (user control) whereas smart wheelchairs can use head gestures (i.e., head tilt), computer vision [25], voice based navigation using a microphone, or Robot Operating Systems (ROS) [26]. Other inputs include hand gestures, tongue movement, and eye-gestures to control the powered wheelchair direction [27]–[29]. Alternate controls are beneficial for people who cannot use the traditional joystick controller [30]. Table 2-1 describes user-input methods implemented in smart wheelchairs.

Table 2-1 Input-control modes implemented in smart wheelchairs.

Input	Description	
Mechanical Joystick [31] [32]	Hand/wrist or chin motion converted to wheelchair navigation commands. Requires cognitive and motor skills. Low computational power requirements.	
Sip-n-puff breath control [33]	Breath control converted to wheelchair navigation. Uses pressure sensors to detect user breath pressure to navigate the wheelchair.	
Camera / Vision control [34]	Head tilt/eye gesture/hand gestures converted to wheelchair navigation commands. Requires high processing power.	
Microphone / Voice [32]	Voice commands converted to wheelchair navigation commands. Requires minimum input from the user (only directional commands).	
Magnetic and bio-signal electrodes [28]	Eye-gestures, tongue gestures, EMG (muscle movements). Requires understanding powered wheelchair control.	

The sensing system consists of obstacle detection sensors (proximity sensors), such as ultrasonic or photoelectric (infra-red or laser) range finders, and are used to identify potholes, curbs, or moving objects that may collide with the powered wheelchair. Each sensor has advantages and disadvantages in terms of cost, accuracy, and robustness in detecting obstacles. Cameras for obstacle detection have been extensively researched in the recent past, since cameras offer a wide detection range and have a small form factor that can be easily mounted in different places on the powered wheelchair [5]. However, cameras require high processing power to extract useful information from the video. Many researchers have combined multiple sensing technologies to detect obstacles, to reduce dependency on unreliable sensors. The control software is the brain behind converting user direction commands and sensor information into wheelchair motor commands. Figure 2.2 [35] describes a simple control system used in smart wheelchairs.

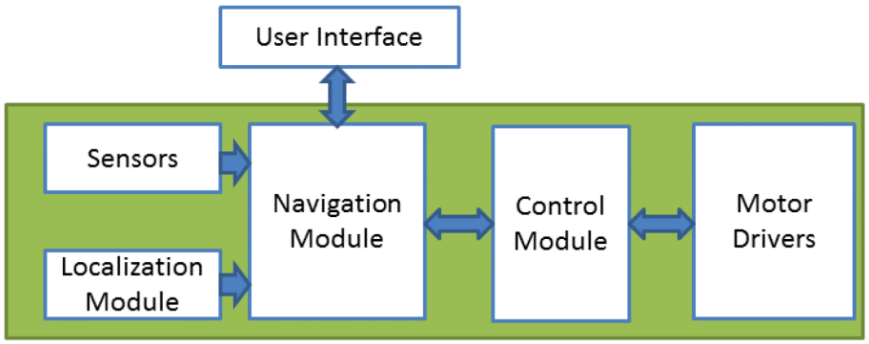


Figure 2.2 Typical smart wheelchair control system [34].

2.5.3 Operating modes

Control software can provide different operating modes to give the user different levels of autonomy. A smart wheelchair with low autonomy gives the user absolute control but will either help the user navigate or help avoid obstacles. This operating mode can be beneficial for people who can effectively plan a path but require navigation assistance. If the user cannot execute a path or effectively control the wheelchair, a smart wheelchair with high autonomy will drive the user to a pre-mapped location by itself. This works best if the person is within a fixed and known environment [30]. Based on functionality and level of autonomy, operating modes can be classified as obstacle avoidance and assistive navigation; mapping, localization, and path planning; and human following.

2.5.4 Obstacle detection/avoidance and assistive navigation

Smart wheelchairs with obstacle detection/avoidance were developed to assist in detecting real-time obstacles around the wheelchair and play an important role in avoiding collisions, thus reducing accidental contact. In Figure 2.3, a mobile robot uses an object anti-collision algorithm to maintain a constant distance from objects when maneuvering toward a final destination. These objects are obstacles, such as walls or furniture, that exist in the mobile robot's global path [23]. Collisions are detected by proximity or range sensors. Photoelectric and ultrasonic sensors are usually mounted under the powered wheelchair carriage, facing outward to create a 360-degree field of view (Figure 2.4) [22].

2.5.5 Sensors for obstacle avoidance

Obstacle detection/avoidance sensors must operate over small sensing regions and with less resolution [36, p. 279] to maintain a controlled sensor field-of-view close to the mobile robot or smart wheelchair, to detect objects in their small field-of-view. The most common sensors for smart wheelchair obstacle avoidance are photoelectric and ultrasound sensors. Other range sensors used in object detection are magnetic, inductive, capacitive, and microwave sensors [36].

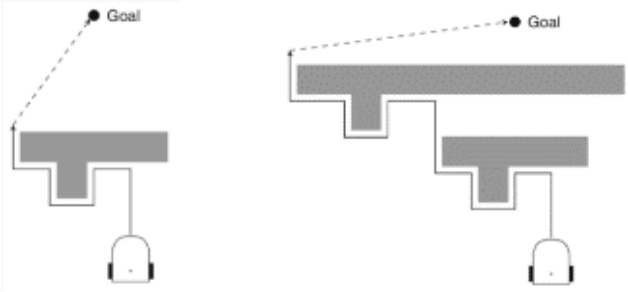


Figure 2.3 Obstacle avoidance system with mobile robot avoiding walls by detecting a solid structure in its global path towards the goal.

Ultrasonic and photoelectric range sensors output an analog or digital signal based on the distance to an obstacle. Proximity sensors work with modulated or pulsed time of flight (ToF) modes (Figure 2.4).

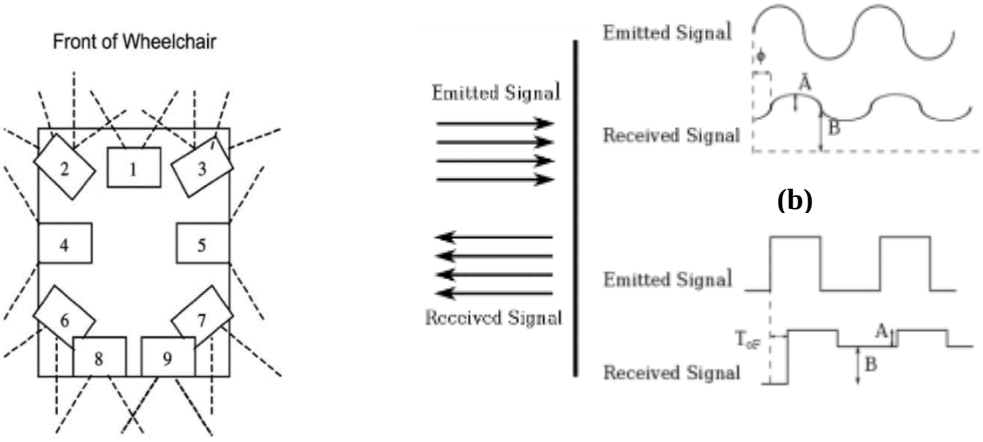


Figure 2.4 (a) Photoelectric and ultrasonic sensors mounted under the wheelchair carriage facing outward (360-degree field of view) [22], (b) Modulated time of flight [37], and (c) Pulsed time of flight [37].

The modulated ToF sensor emits a continuous time modulated sinusoidal signal. Distance is calculated by measuring the phase difference between the emitted and received signals using equation 2.1 [37],

$$L = \frac{x\phi}{2\omega} = \frac{x\phi T}{4\pi} \quad \textbf{(2.1)}$$

where x is the speed of sound in air (343 m/s) or speed of light ($3*10^8$ m/s), based on the type of sensor, ω is the angular frequency, T is the signal's period, and ϕ is the phase difference. Pulse ToF sensors emit pulsed square wave signals. Distance is calculated by measuring the time difference between signal emission and its return to the sensor, after being reflected by an object [37], [38], and is given by equation 2.2,

$$L = \frac{c * ToF}{2} \quad \textbf{(2.2)}$$

where c is the speed of sound in air (343 m/s) or speed of light (3*108 m/s) based on the type of sensor, ToF is the signal's time-of-flight from emission to reflection from the sensor, and L is the distance between the sensor and the object. Table 2-2 describes advantages and disadvantages of ToF range sensors for obstacle detection/avoidance [36].

Table 2-2 Advantages and disadvantages of time-of-flight range sensors.

Advantages	Disadvantages
- High linear range accuracy due to line-of-sight sensing (emitted and reflected signals propagate in the same path) - Output distance information is directly proportional to the strength and time-difference of the signal, eliminating complex computation analyses - High signal resolution in smaller distances - Target material properties (reflectiveness, texture, color, spatial orientation) have little to no effect - Small form-factor	- Reduced sensing capability due to line-of-sight - Timing uncertainties due to the received signal's dynamic range, caused by varying reflectiveness of the target surface, causing ToF detection errors - For ultrasonic sensors, speed of sound is greatly influenced by atmospheric conditions such as temperature and humidity, creating ToF detection errors - For photoelectric sensors, target material properties (reflectiveness, texture, color, spatial orientation) have a major effect.

Researchers have employed ultrasonic and photoelectric sensors, or a combination of the two, for smart wheelchairs obstacle detection/avoidance. The Robotic Intelligent Wheelchair System (RIWS) used a combination of sonar and infrared sensors mounted at the powered wheelchair base for obstacle detection and avoidance. The sensors were mounted perpendicular to each other to increase the detection area. Using multiple types of sensors reduced the error caused by uncertain sensor readings and increased system redundancy [39].

Most obstacle detection/avoidance systems are integrated with an assistive navigation system, an algorithm designed to assist the user in navigating the powered wheelchair based on user commands and information from the obstacle detection/avoidance systems. Two operating modes are considered, each with a varying degree of user control: minor navigation assistance and collision avoidance, or major path re-routing capabilities that require little control from the wheelchair user. For example, the NavChair was developed to provide obstacle avoidance with assistive navigation control for users with varying assistive needs [21]. The obstacle avoidance system consisted of twelve ultrasonic transducers facing the wheelchair's front and two avoidance algorithms, vector force field (VFF) and minimum vector field histogram (MVFH). The VFF algorithm was developed for irregularly shaped mobile robots and works by modifying the direction and speed by detecting every obstacle's distance from the wheelchair on the current path. MVFH uses distance data from sonar sensors and wheel motion to update a Cartesian map (certainty grid) around the wheelchair. The wheelchair direction is modified based on the probability of an obstacle existing within a grid. The NavChair's assistive navigation also includes 'door passage' mode for moving the wheelchair through a door and automatic wall following that uses sonar sensors to follow a wall [21].

A smart wheelchair with obstacle avoidance based on multi-sensor fusion was developed to assist users in powered wheelchair navigation [39]. Four ultrasonic sensors, a camera, and a digital signal processor based on an ARM9 embedded system (DSP) were used to determine and avoid obstacles. The obstacle avoidance algorithm was based on a fuzzy neural network that used signals from ultrasonic sensors and images from the camera to determine the object's position and shape. The wheelchair speed and direction were modified based on the position and distance of an obstacle to the wheelchair's center.

Obstacle detection/avoidance and assistive navigation were researched and implemented in mobile robots and smart wheelchairs with improved performance for people with varying

disabilities. Autonomous navigation and assistive technology may be beneficial for users who have trouble navigating and obstacle detection/avoidance may be beneficial for those who are unsafe drivers [14].

2.5.6 Mapping, localization, and path planning

In robotics, localization and mapping is used to estimate the current state of a mobile robot, using information from sensors to construct a model (local map) of the environment [40]. The mobile robot's state comprises the pose (i.e., position, orientation), robot velocity, and map (the environment) [41]. The first step in autonomous navigation is to determine the surrounding map and calculate the mobile robot's local pose and robot velocity. The most common localization and mapping method is Simultaneous Localization and Mapping (SLAM) that uses recursive probabilistic models for simultaneous estimation of the mobile robot state by using exteroceptive sensors such as ultrasonic, photoelectric range finders, cameras, and proprioceptive sensors such as wheel encoders, inertial measurement units (IMU), or global positioning systems (GPS) [40], [42]. The major objective of SLAM is to map the surrounding environment and determine the real-time pose and velocity of a system, when neither are known before-hand. This objective is achieved with a mathematical multi-modal approach that includes the robot motion model, a direct observation model, and an inverse observation model.

The motion model is the first step in the SLAM algorithm, navigating a mobile robot from one pose to the next. Assuming that a mobile robot 'R' navigates in a plane with its current pose as the position and orientation matrix, and navigates to the next point according to a direction instruction 'u' set by a controller in the form of a linear and angular pose vector with a perturbation 'n' caused by error, the robot's new pose can be expressed as,

$$R = f(R, u, n) \tag{2.3}$$

where R and u are given as (where, x, y, θ is the robot's position and orientation),

$$R = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \qquad u = \begin{bmatrix} \delta x \\ \delta y \\ \delta\theta \end{bmatrix}$$

The motion model is a mathematical model for increased localization uncertainty when the robot navigates from the initial position to the next. Uncertainty increases due to increased unavoidable noise and error in gathering and calculating position. The direct observation model

determines position and orientation of features or landmarks in the environment using exteroceptive sensors, which need to be incorporated into the map. A high degree of uncertainty exists in direct observation due to noise and errors caused by unreliable exteroceptive sensors. An inverse observation model is used to reduce this uncertainty by predicting robot and landmark poses based on the motion and direct observation models of the previously observed map.

Although high computing power is required, SLAM is safe and accurate for smart wheelchair navigation in unknown indoor environments [30], [42]. For example, the SLAM based cross-a-door robotic wheelchair used an Extended Kalman Filter (EKF) to navigate through a door [42]. System architecture included a brain-computer interface, door-crossing path planner algorithm, and trajectory controller. The EKF-based SLAM algorithm determined the wheelchair's pose in the environment using a Laser Range Sensor (LRS) and extracted lines and corners to define the doorway. The doorway central pose was represented as a point in Cartesian coordinate space. The path planning and mapping algorithm created dynamic nodes between the current wheelchair pose and the doorway central pose, and the trajectory controller navigated the wheelchair by satisfying the closest node to the wheelchair. Once the wheelchair navigated to the first node, the EKF-based SLAM algorithm re-estimated poses and a new path towards the doorway was calculated, completing doorway crossing by re-iterating the procedure. Figure 2.5 shows the door-crossing path planner algorithm and node generation for three situations [42].

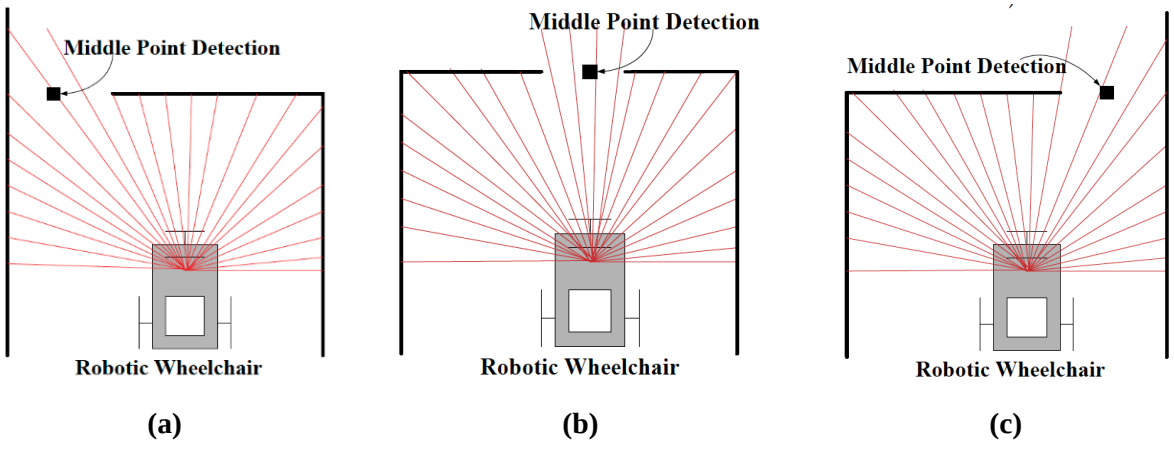


Figure 2.5 Door-crossing detection algorithm: (a) Door-way is left of wheelchair, (b) Door-way center of wheelchair, and (c) Door-way right of wheelchair [42].

Usually, localization and mapping algorithms are incorporated with path planning and trajectory controller algorithms to navigate a mobile robot/powered wheelchair towards a desired location (goal). Path planning algorithms are developed to determine the overall optimum path required for the robot to safely navigate in static environments; such as, furniture, walls, or rooms [41]. Path planning algorithms usually consist of straight paths and sharp curved turns between waypoints generated by the algorithm (Figure 2.6), which is not desirable for smart wheelchair navigation because straight paths and sharp turns may cause injury to people or objects on and around the device. Therefore, path planning algorithms incorporate path smoothing to achieve a desired smooth path from the current position to the destination (Figure 2.6).

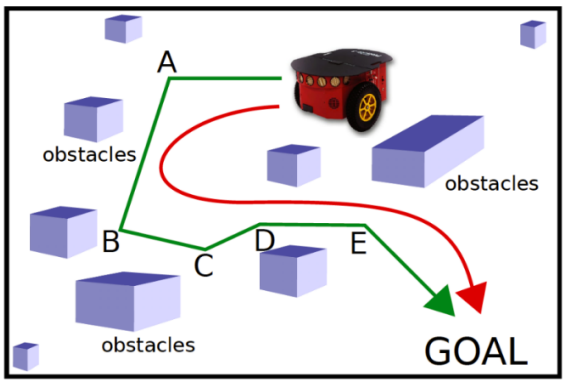


Figure 2.6 Generalized path planner algorithm. The green path consists of straight lines and sharp turns at A, B, C, D, and E. The green path is generated by the path planning algorithm. The smooth path generated by the path smoothing algorithm is in red [41].

2.5.7 Path planning algorithms

Path planning algorithms can be categorized into interpolation based path smoothing, path smoothing using special curves, and optimization-based path smoothing [41]. Polynomial based interpolation calculates a suitable smooth path around obstacles. This literature review only covers the most commonly used path planning algorithm for smart wheelchairs: Lagrange interpolation smoothing algorithm [41],

$$P(x) = \sum_{j=1}^{n} P_j(x) \tag{2.4}$$

$$P_j(x) = y_j \prod_{k=1, k\neq j}^{n} \frac{x - x_k}{x_j - x_k} \quad \textbf{(2.5)}$$

where (x_j, y_j) are pairs of nodes that exist in the path and the function $P(x)$ returns interpolated points around the nodes, thus giving a smooth curve. A trade-off exists between a better straight-line path fit, a smooth curved fit, and a computationally high interpolating function due to highly smooth curves requiring more data points, thus resulting in a higher degree polynomial that can produce an unstable algorithm. Therefore, as a disadvantage, higher-degree polynomial interpolation may generate accurate data points but generate oscillatory paths between waypoints.

Development and integration of mapping, localization, and path planning systems for powered wheelchairs is challenging because of unreliable sensor readings, especially in dynamic outdoor environments. Therefore, researchers have used an array of multiple exteroceptive sensors, such as LRS and cameras, to increase localization and mapping algorithm accuracy. Kalman Filters can reduce uncertain sensor readings from proprioceptive sensors, such as wheel encoders, inertial measurement units (IMU), or global positioning systems (GPS). However, in addition to a computationally high path planning algorithm, multiple sensors greatly reduce efficiency and increase smart wheelchair cost. Nevertheless, these algorithms are advantageous for safe navigation in situations where the user needs assistance; such as, narrow paths, doorways, footpath, or narrow corridors [30].

2.5.8 Human following (tethering)

Several smart wheelchairs have been developed to follow alongside an object or human (companion or a caregiver) using technologies adapted from mobile robot object detection, localization, mapping, and path planning. Human following systems consist of a complex object pose and velocity detection algorithm integrated with a trajectory controller. The pose and velocity information extracted from the object/human determines the wheelchair trajectory, thereby tethering the wheelchair to the person or an object [43]–[45]. Researchers have used different technologies to determine the person's pose and velocity; such as, ping sensors, photoelectric sensors, cameras, or variable tension strings. These techniques can be grouped into wired/contact tethering or wireless/contactless tethering.

Wired or contact tethering uses a physical or a mechanical connection between the mobile robot/wheelchair and the accompanying person. In the past, researchers have developed and implemented wired tethering in mobile robots to move heavy loads. For example, tether steering was used to carry luggage or as a jogging companion in crowded (dynamic) environments. Tethered steering uses a mechanical string tether (Figure 2.7) between a mobile robot and a person and measures the string tension and angle to calculate the person's trajectory and determine the mobile robot's direction and velocity.

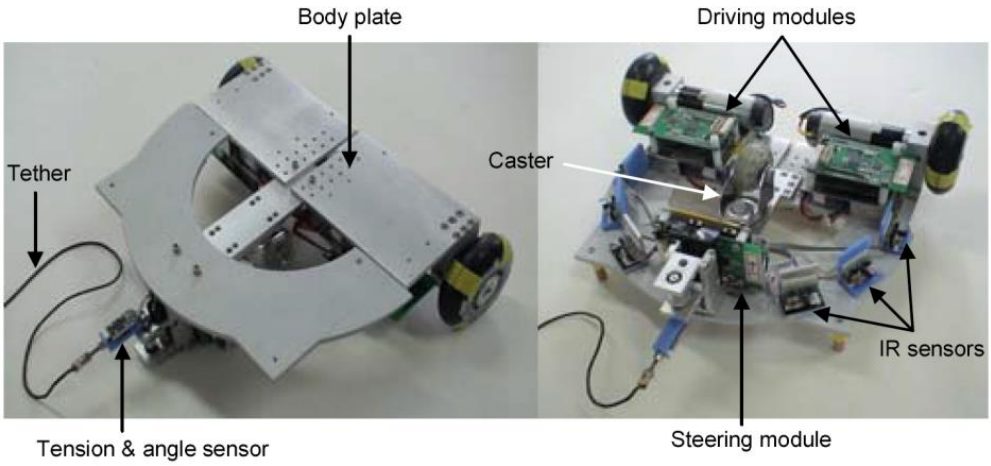
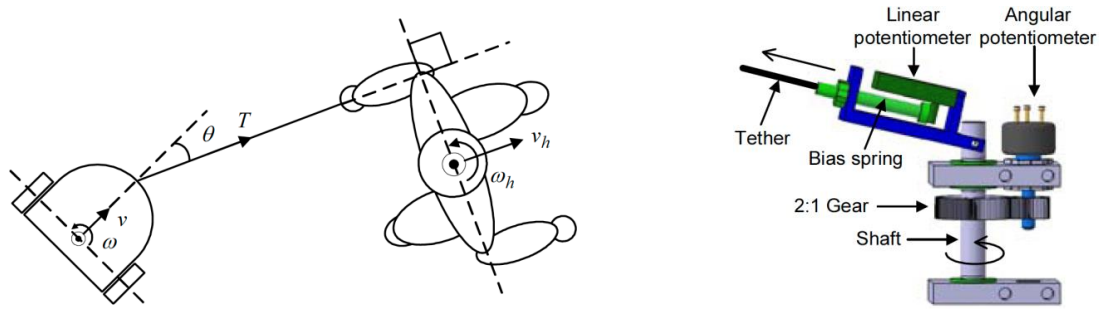


Figure 2.7 Wired/contact human tethering [43].

The mobile robot followed the user from the back, with the person holding a tether string (Figure 2.8). The physical tether between the person and robot used linear and angular potentiometers to sense string tension and angle. Robot trajectory was calculated from two adaptive motion control algorithms, one to determine the robot's pose and the second to determine the human's pose. The linear and angular potentiometer data were used to calculate a reference trajectory using the robot's motion control algorithm. The mobile robot's velocity was modified by calculating the user's velocity, acceleration/deceleration, and direction, thereby providing smooth navigation behind the user. The prototype was tested in a 3-D motion-analysis laboratory to determine the mobile robot's performance [43].



(a) **(b)**

Figure 2.8 (a) Wired/contact human tethering between a human and robot with tether length, orientation angles, and positions, and (b) Prototype for tether tension and angle [42].

The Navi-guider is an add-on module to mobile robots, using a reed and reel mechanism to calculate a mobile robot's speed and direction for following. A tether string was used to determine the person's pose and velocity. The string could be pulled from the mobile robot to a maximum length of 210 cm. A mechanical system measured the tether length using the reel mechanism and direction of pull using the reed mechanism. The direction and velocity of pull, which determined the user's pose and velocity, were determined from potentiometer signals and an encoder connected to reel mechanisms. Robot trajectory was controlled using two motors. Velocities for each motor were given by equation 2.6 and equation 2.7:

$$\begin{aligned} V_L &= K_l L \\ V_R &= K_l L - K_2|\theta| \end{aligned} \qquad for\ \theta < 0 \tag{2.6}$$

$$\begin{aligned} V_L &= K_1 L - K_2\theta \\ V_R &= K_1 L \end{aligned} \qquad for\ \theta \geq 0 \tag{2.7}$$

where V_L and V_R were left and right motor velocities, $K_1 L$ was the nominal velocity (K_1 = Gain, L = tether length), and $K_2\theta$ were additional velocities at each motor (K_2 = Gain, θ = Orientation angle (rad)) [46]. A simple Proportional-Integral-Derivative (PID) controller was used for smooth velocity control. The prototype was tested in a controlled environment and trajectory data was collected using a localization sensor. The Navi-guider was allowed to follow a commercial Pioneer robot (ActiveMedia Robotics). Unfortunately, the short

mechanical tether hindered the person's free movement and made control around corners difficult, especially in dynamic environments. The prototypes described above are designed only for mobile robots that navigate behind the user and not side-by-side or in front following, which would be beneficial for powered wheelchair social-following.

Wireless/contactless tethering would be advantageous over wired/contact tethering in crowded environments. Cameras, ultrasonic, or photoelectric range sensors could be used to determine the person's or object's pose and velocity. Sensor information is usually processed by advanced processors to determine the person's and powered wheelchair velocity and trajectory. Wireless tethering research for human following in smart wheelchairs can use cameras, laser, or ping sensors.

A robotic wheelchair with omni-directional vision [46] is the only tethered device to offer side-by-side following. This standalone wheelchair had two goals, ease the burden on the caregiver and initiate easy communication between the caregiver and the powered wheelchair user. This wheelchair had four LRS, micro-computer, laptop computer and touch sensitive interface. Three LRS were mounted at the bottom for obstacle detection and a fourth sensor was mounted on a vertical pole behind the powered wheelchair, with a 360 degree field of view omni-directional camera for caregiver detection (Figure 2.9). The LRS output a signal proportional to the caregiver's distance from the sensor. This sensor information was used to calculate the caregiver's pose with respect to the powered wheelchair.

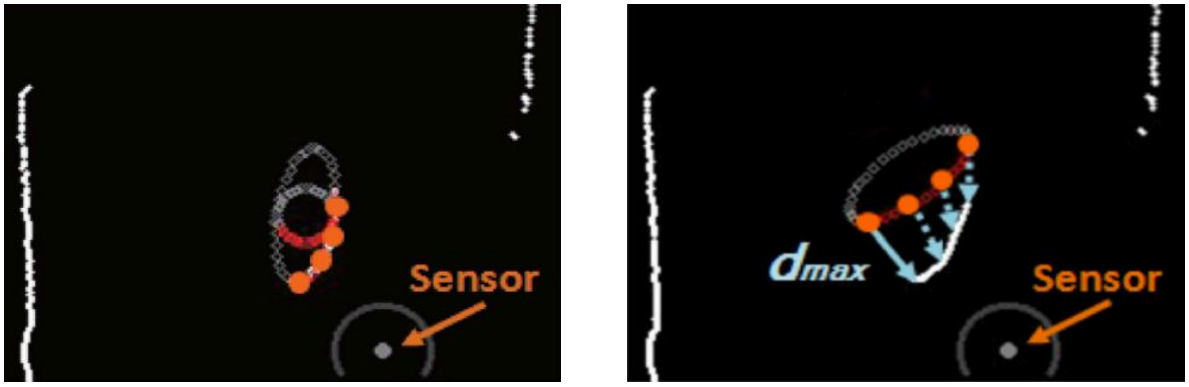
(a) **(b)**

Figure 2.9 (a) Robotic wheelchair prototype for moving alongside a caregiver, (b) Omni-directional camera and an LRS attached to the wheelchair's back [47].

The detection and following algorithms can be categorized into caregiver tracking and caregiver following. Caregiver tracking is the process of identifying and determining the caregiver's pose. The LRS and omni-directional camera were mounted at a height that matched the average human shoulder height from the ground. Caregiver identification used the omni-directional camera and initial caregiver position was first entered on a touchscreen and the system stored the position. The omni-directional camera was advantageous over the LRS because it was easier to visually identify the caregiver when following in a crowded environment. The caregiver's shoulders were modeled as an ellipse using the LRS data, represented as the ellipse center (u, v) and rotation angle ϕ, and projected onto a 2-D laser image (Figure 2.10). The likelihood of a caregiver around the detected contour was calculated and the system positively tracked the caregiver based on the result of equation 2.8,

$$w_{t,laser}^{(i)} = exp\left(\frac{-d_{max}^2}{\sigma_d}\right) \quad \textbf{(2.8)}$$

where $w_{t,laser}^{(i)}$ is the likelihood score based on the laser image, σ_d is the variance of the distances from each point on the current contour at every time instance (t), and d_{max} is the maximum distance between the reference contour and the current position (Figure 2.10).



(a) **(b)**

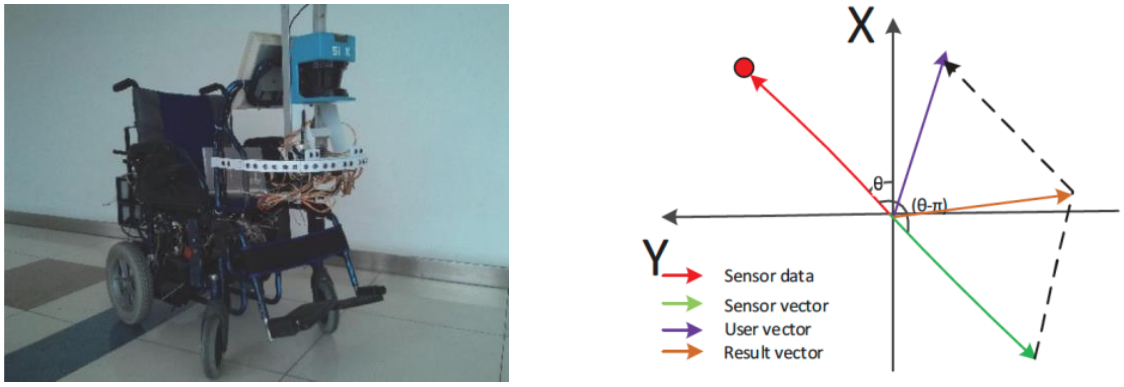
Figure 2.10 (a) Caregiver shoulder contour modeled as coordinate points in 2D space, and (b) caregiver tracking using the maximum distance between shoulder contours [47].

The smart wheelchair speed and direction were controlled by the caregiver's pose. The initial position, entered on the touchscreen, was used to calculate the fixed tether distance between the powered wheelchair and the caregiver. At every instance, the system calculated new caregiver positions and orientations (u_1, u_1) and determined the required wheelchair speed (s) and direction (ϕ_1) by comparing and subtracting the new position from the initial position. If this difference was positive ($u_0 = u_1 - u$; $v_0 = v_1 - v$; $\phi_0 = \phi - \phi_1$), the wheelchair increased speed and turned left. If the difference was negative, the wheelchair reduced speed and turned right. If there was no difference, the speed and direction were constant.

Obstacle avoidance and path re-routing algorithms assisted in navigation during situations where caregiver following was not desirable. To avoid obstacles or pedestrians and reduce accidents, the algorithm changed from a following mode to follow-back mode. Semi-autonomous caregiver-following was advantageous for wheelchair users who are usually accompanied by caregivers. Testing methods and results were not disclosed.

A smart wheelchair with driver assistance and human following [47] was designed to increase powered wheelchair user safety (driver assistance) and improve social activities and interactions. For people with some locomotor capacity, human following mode could enable the user to walk around and still have access to the wheelchair when needed. The smart wheelchair system was implemented on a JiaoLong Wheelchair, with wheel encoders, ultrasonic sensor ring mounted around the wheelchair (Figure 2.11), and a digital signal processor (DSP) that provided closed-loop control.

The system used a shared control technique with joystick input and environment information from ultrasonic sensors to determine suitable wheelchair motion commands. The driver assistance system's rule-based algorithm used ultrasonic sensor data to determine if an obstacle was in the sensor field of view. A Virtual Field Force (VFF) technique calculated the probability of an obstacle coming in contact with the wheelchair in its current trajectory. The system generated a vector based on sensor data and the user joystick input (Figure 2.11).



(a) **(b)**

Figure 2.11 (a) JiaoLong wheelchair with driver assistance and human following [48], and (b) Motion vector generation based on the direction of obstacles and person to follow [47].

Human following mode used a target (person) tracking and target motion prediction algorithm. A transmitter on the person walking in front, ultrasonic sensor ring, and triangulation function determined the approximate user position, when the user was in the ultrasonic field-of-view. The ultrasonic sensors output a signal proportional to the user distance using ToF. The triangulation function used a time synchronized signal generated by the transmitter to calculate the user orientation with respect to the wheelchair, and was given by,

$$\theta_T(t_k) = arctan\left(\frac{y_T(t_k) - y_T(t_{k-1})}{x_T(t_k) - x_T(t_{k-1})}\right) \quad \textbf{(2.9)}$$

where θ_T is the target (user) orientation, x_T, y_T is the target position and t_k represents the k^{th} iteration of time t. When the user cannot be found in the ultrasonic sensor field-of-view, the system used odometry data and previously known positions of both the target and the wheelchair to predict the relative target orientation (dead-reckoning navigation). The authors conducted a test to measure smart wheelchair usability during daily activities; such as, crossing a door, human avoidance, wall avoidance, stairs detection, backing up, and non-obstacle driving. The authors concluded that the system benefited users who need driver assistance while crossing narrow doors, avoid obstacles, and human following. The authors did not present test results.

Guide following smart wheelchair control, using laser range sensor [48], provided additional navigational assistance by following behind a guide. A powered wheelchair was equipped with left and right wheel resolvers (analog-encoders), yaw rate gyroscope, and LRS. Wheel encoders were mounted directly on each wheel. The gyroscope was mounted on the wheelchair backrest side and LRS was mounted at the wheelchair's front (Figure 2.12).

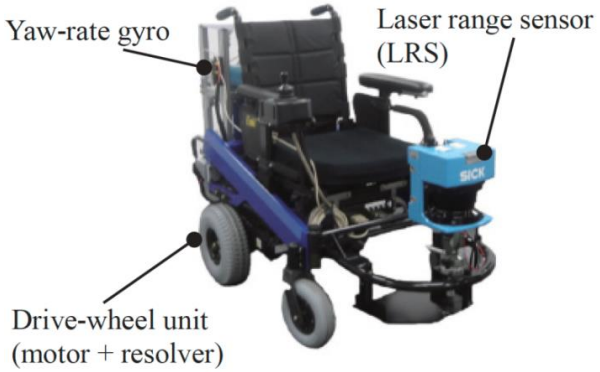
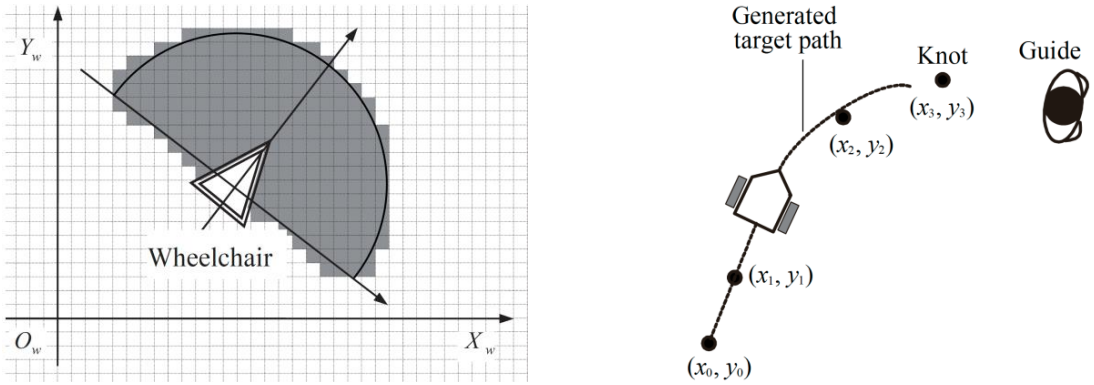


Figure 2.12 Guide following smart wheelchair prototype using a laser range sensor [48].

The person following control system can be classified as person detection, path generation, and person following control. Person detection used a reflective surface for identification, with the person showing the reflective surface towards the LRS to switch between normal user-control and following modes. The algorithm used LRS information to fill an occupancy grid (Figure 2.13) at every time instance and marked each grid if an object was found. The reflective surface would be registered in one of the occupancy grids.

The algorithm generated a threshold circle around the marked grid and used the LRS data to determine if a moving object was within the threshold region. If a moving object was identified, the system positively identified the person to follow. The person's position in the occupancy grid was calculated by using a Kalman filter based on LRS information.

The path generation algorithm navigated the wheelchair towards the person by generating knots or waypoints between the wheelchair and the person, based on LRS, wheel resolvers, and the gyroscope. Wheelchair position was calculated using dead-reckoning, which used the resolvers and gyroscope to determine the wheelchair's current position based on previously known positions, which could be previously generated waypoints. Each waypoint was represented as a point in coordinate space (Figure 2.13), where waypoint A was the current wheelchair position.



(a) **(b)**

Figure 2.13 (a) Grid map generated by the control system (LRS field of view as gray cells in a coordinate plane), and (b) Waypoint and path generation based on guide position [48].

The difference between each waypoint was fixed to 0.7 m. A path was generated based on the known waypoints using a 3rd order cubic spline interpolation technique, which can be expressed using equation 2.10,

$$x = f(\xi) = (a_0 + a_1) + a_1\xi + b_0N_0(\xi) + b_1N_1(\xi) \quad \textbf{(2.10)}$$

where $(0, x_0)$, $(1, x_1)$, and $(2, x_2)$ were the first three waypoints and $(3, x_3)$ was the unknown fourth waypoint based on person's new position; a_0, a_1, b_0 and b_1 were function coefficients; and $N_0(\xi)$ was the difference quotient of the three waypoints, a_0, a_1, b_0 and b_1. While moving towards the calculated waypoint, a new way would be generated based on the person's new pose. Equation 2.10 was repeated for every new waypoint determined by the LRS.

Each time, the first waypoint $(0, x_0)$ was the wheelchair position and the last waypoint $(3, x_3)$ was the person's current position. The person following algorithm determined wheelchair velocity and orientation based on the constructed path. Wheelchair velocity was calculated by a feed-forward and PI feedback controller to maintain a constant distance from the person:

$$v^* = \frac{1}{cos\alpha}\left\{v_t + K_1(l_t - l^*) + K_2\sum_{t=1}^{t}(l_t - l^*)\right\} \quad \textbf{(2.11)}$$

where α is the orientation angle with respect to the person; v_t is the person's current velocity, determined by comparing the person's speed expressed in the occupancy grid using the Kalman filter; K_1 and K_2 are the controller (control) gains; l^* is the required gap between the wheelchair and the person; l_t is the current length between the wheelchair and the person. By maintaining a constant $(l_t - l^*)$, the wheelchair maintained a constant distance behind the person. Wheelchair orientation velocity (turning velocity) with respect to the person was calculated using the wheelchair kinematic model, by calculating the wheelchair's state at each waypoint (using the left and right encoders and gyroscope). The individual wheelchair motor velocities were expressed by calculating inverse kinematics related to the powered wheelchair and using the wheelchair tread distance (L) and velocities at each motor using equation 2.12,

$$\begin{pmatrix} v^*_{left} \\ v^*_{right} \end{pmatrix} = \begin{pmatrix} 1 & -\frac{1}{L} \\ 1 & \frac{L}{2} \end{pmatrix} \begin{pmatrix} v^* \\ \theta^* \end{pmatrix} \quad \textbf{(2.12)}$$

The prototype described in the paper [49] was designed to follow a person from the back using an LRS. The prototype was tested by following a person moving at 0.21 m/s, with 2 m between the person and the wheelchair. The system had a maximum path tracking error of ±0.1 m and a gap control error of ±0.25 m, when following the person from the back. Further research is needed to develop a guide-follower wheelchair in crowded environments.

2.5.9 Human following in mobile robots

A human following mobile robot [43] was developed as an assistive-intelligent cargo transportation system. The robot followed from the back using LRS to detect the person's pose. LRS was mounted on a commercially available Kobuki (Yujin Robot, South Korea [50]) at 25 centimeters from the floor and connected to a laptop computer running ROS (Figure 2.14).

The LRS detected the person's shin by observing a contour that represented the shin's posterior shape. Figure 2.15 shows the contour in 2-D space (top-view) for four human orientation angles. The shin contour information was described as two parabolas, one for each shin. The shin's parabolic geometric characteristics (coordinate position in 2-D space, angles, shape) were used to calculate shin clusters and then determine the person's pose in a search area in front of the mobile robot. The human following robot used an algorithm that controlled the speed and direction of the mobile robot based on the human's speed and direction of motion.

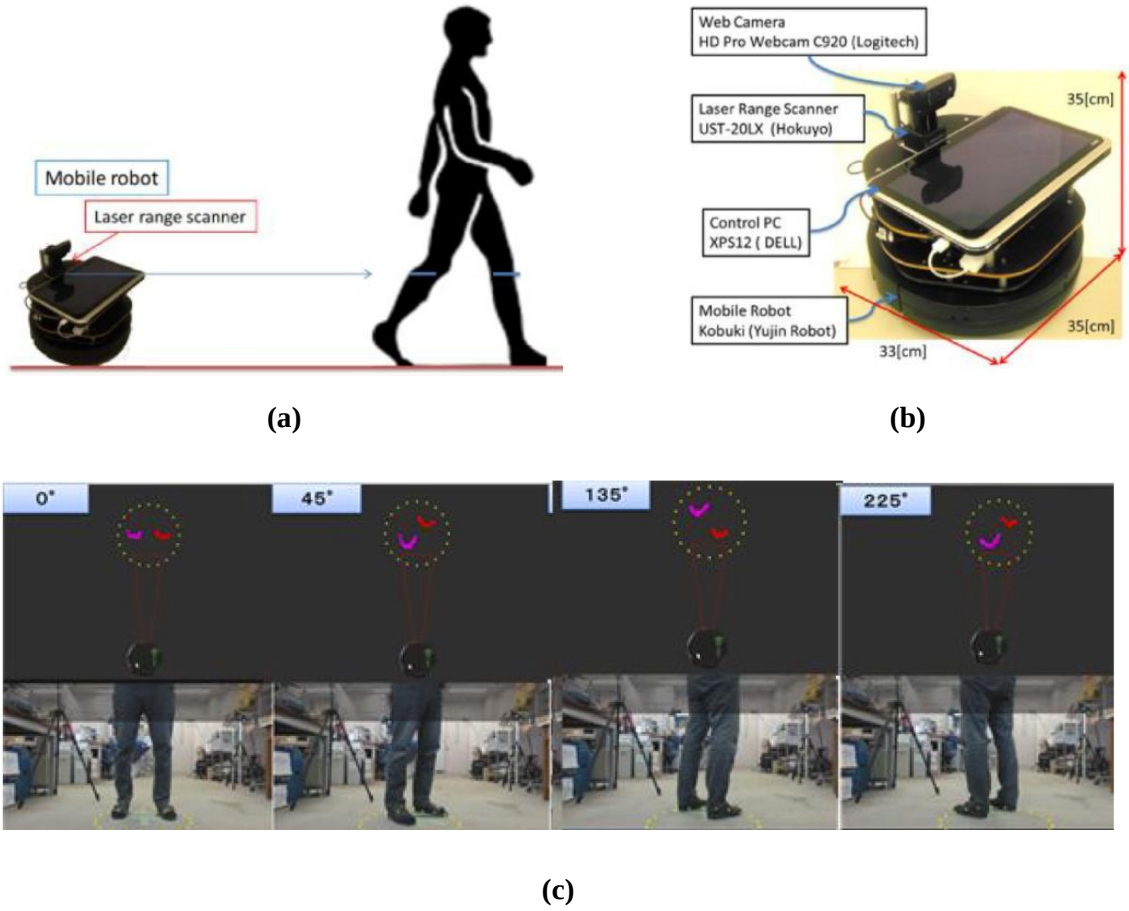


Figure 2.14 (a) Human following mobile robot based on LRS, (b) Human following mobile robot with web camera, LRS, and laptop computer [44], and (c) LRS scanning of the person's shin for different person orientation angles [43].

Speed was determined by calculating the offset distance between the reference center of both parabolas within the search area and the actual parabolic positions. Direction was calculated by measuring the offset angle between the reference center of both the parabolas within the search area and the actual parabolic positions. The mobile robot moved towards the target (final position) based on the calculated speed and direction. A collision avoidance algorithm was implemented using LRS, to avoid obstacles in the path between the mobile robot and the person. The algorithm identified an object in the vicinity and created a buffer area (represented as a circle in Figure 2.15) around the object, based on the mobile robot's size. The robot created a tangential path along the circle in the person's direction, thus avoiding the obstacle. The authors conducted two performance

tests. The first test measured human following in a rectangular path and the second test follow a person along a pre-determined path in a room. The first test resulted in an 83% success rate for identifying and determining the person's pose. Error occurred when the LRS did not identifying the person's shins along sharp right-angled turns. The second test resulted in a 66% success rate, with errors caused by misrecognizing a table's leg instead of the person's shins.

2.6 Summary

This literature review includes a description of existing powered wheelchair user demographics, benefits of using a powered wheelchair and the types of accidents caused by powered wheelchairs, and smart wheelchair technologies inspired from mobile robots that increase wheelchair user's safety during powered wheelchair operation. The literature shows promising and innovative smart wheelchair technologies for assistive navigation, autonomous control, and human following. These wheelchairs employ a plethora of input sensors, controllers, and algorithms to assist users in wheelchair navigation, thereby aiming to reduce powered wheelchair accidents such as tipping and falling, accidental contact, and dangerous operations. Table 2-3 summarizes human-following smart wheelchair technologies.

The human-following techniques in Table 2-3 employ follow-behind or side-by-side following. Currently, human-following techniques employ computationally high algorithms that require bulky and expensive sensors and processors to process data and calculate the person's and wheelchair's position, thereby calculating the required trajectory based on the person's path.

The purpose of this thesis is to develop an inexpensive powered wheelchair add-on smart unit for human-following, especially for users who require a device that reduces the burden of controlling the powered wheelchair during social situations, by enabling side-by-side following. Although a number of researchers have implemented human-following in smart wheelchairs, only one project targeted a social following technique (side-by-side-following), and this caregiver following approach required extensive modifications to the underlying wheelchair [47]. This thesis aims to provide a cost effective yet, a robust system that can be used by any commercially available powered wheelchair without modifications to the underlying powered wheelchair.

Table 2-3 Human following technologies implemented in wheelchairs.

Study	Sensors	Description	Advantages	Disadvantages
Following Wheelchair, Japan [47]	Omni-directional camera, Laser range sensors (LRS)	Omni-directional camera and LRS detect and determine person's pose. Person tracking using likelihood method. Motion detection using person's shoulder orientation. Person identification using omni-direction camera. Pose detection using LRS	Continuous person tracking. Possible side-by-side following	Sensors are affected by environmental conditions (light, reflective surfaces). Pose estimation based on studying person's shoulder orientation (high uncertainty)
Driver assistance and human following smart wheelchair, China [48]	LRS, Ultrasonic range sensors, encoders	Driver assistance and human-following. LRS and ultrasonic range sensors detect and determine person's pose (transmitter for identification, triangulation for pose)	Shared control between wheelchair user and control system is advantageous for users who require additional navigational support due to lack of cognition	Human pose prediction is erroneous due to uncertain signals from 18 ultrasonic sensors. Potential uncertainty in identifying person due to motion prediction algorithm errors
Guide follower smart wheelchair, Japan [49]	LRS, wheel resolvers, gyroscope	Follows person from behind. LRS determines person's pose. Wheel resolvers and gyroscope determine wheelchair position using dead-reckoning	Robust navigation using waypoint generation algorithm	Person-wheelchair distance has high error due to fixed waypoints
Vision-based follower, Japan [29]	Stereo Camera	Vision-based human-following. Stereo camera mounted facing forward. Follow-behind. Image processing identifies and determines target's pose	Single stereo camera can determine person's pose, reducing need for additional sensors. Function detects and tracks person in real-time.	Human tracking in crowded environments is not possible since wheelchair could follow any other moving object within the field of view
Fuzzy-based follower, Taiwan [51]	LRS and Camera	Fuzzy-based human following using LRS and camera to identify person to follow and detect person's pose. Sensor-fusion fuzzy logic algorithm identifies target and finds position and orientation	Camera and LRS make the system very bulky and expensive. Facial recognition for target identification could be beneficial for identifying in crowded environments	Facial recognition for target identification increases uncertainty as the person faces away from the camera

3 Modelling and Simulation

3.1 Overview

This chapter discusses the modelling and simulation of an ultrasonic tethering system and aids in addressing objective 1. The ultrasonic tethering system was modelled and simulated in Matlab and Simulink to study the process of determining the AP pose using ultrasonic sensors. The paper also studies wheelchair direction control signals produced by calculating the AP pose using the proposed ultrasonic tethering system.

This chapter consists of a modified conference paper:

> Pingali T, Lemaire ED, Baddour N. Modelling and Simulation of an Ultrasonic Tethering Smart Wheelchair System for Social Following. 42nd Canadian Medical and Biological Engineering Conference (CMBEC42), vol 42, May (2019), Ottawa.

3.2 Abstract

Distracted navigation causes 20% of all powered wheelchair accidents. In social situations, wheelchair users must divide their attention between navigating the chair and conversing with an accompanying person. These conversations could lead to increased mental stress and distractions from maneuvering the chair. This project aims to eliminate the need to manually control a powered wheelchair when moving and conversing with an accompanying person, by controlling the wheelchair's path to follow beside a person. This includes identifying and determining the person's pose to control wheelchair navigation. The proposed ultrasonic tethering system was developed and simulated on Matlab and Simulink using models for ultrasonic sensors, amplification and filtering circuits, and a processing unit. Unlike infra-red sensors and cameras that are highly dependent on environmental light conditions, ultrasonic sensors are inexpensive and independent of environmental conditions. Simulation results determined wheelchair direction based on the accompanying person's pose, suggesting that ultrasonic tethering can be used for side-by-side following. The simulation results can be used to determine circuit component parameters for developing an ultrasonic tethering prototype.

3.3 Introduction

Powered wheelchair accidents may affect user mobility and can result in restrictions in activity and social participation [16] [17]. Accidents such as tipping and falling, accidental contact, and dangerous operations could be caused by distracted navigation of powered wheelchairs [18]. In social situations, powered wheelchair users must divide their attention between navigating the chair and conversing with an AP, thus causing distracted driving. As a solution that minimizes wheelchair control, a smart-wheelchair system that incorporates contactless tethering techniques from human-following mobile robots was implemented on Matlab & Simulink. As explained by the authors in [44], [47], [48], human-following may be beneficial for powered wheelchair users in situations such as:

- Hands-free wheelchair control while moving behind a caregiver/guide
- Participating in social interactions (i.e., conversations)
- Users needing immediate access to powered wheelchair, when walking or exercising.

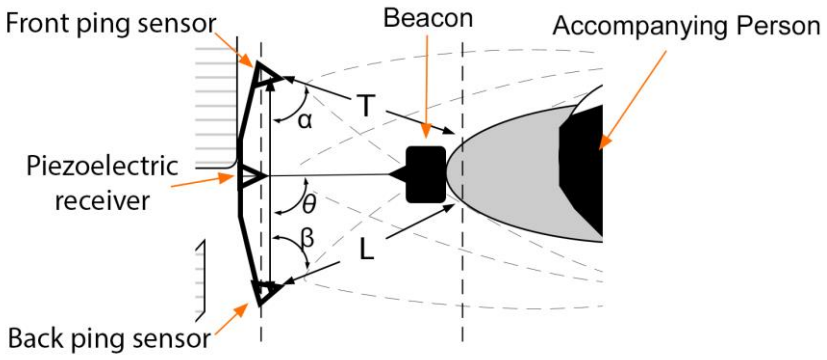
Previous human-following wheelchairs followed the person from behind [44], [47], [48]. However, for comfortable conversations between the user and the AP, the wheelchair should follow the person from the side [52]. Human-following or tethering can be achieved using commercially available infrared range sensors, ping sensors, cameras, or Lidar. Ultrasonic sensors output a voltage proportional to distance by using high-frequency pulses [38] that are not attenuated by environmental factors, such as humidity, dust, and light conditions [53]. The output voltage is stable for shorter object distances, unlike photoelectric sensors [54]. Therefore, ultrasonic sensors were chosen for the proposed tethering system.

This research simulated the proposed ultrasonic tethering approach for social-following. The model was developed on Simulink® Version 9.2 (R2018b) using the Simscape electrical toolbox to determine if ultrasonic tethering is a viable option for side-by-side following. The proposed tethering system consisted of three processes: AP identification, AP pose determination using triangulation, and wheelchair navigation and control using thresholds [44], [47], [48].

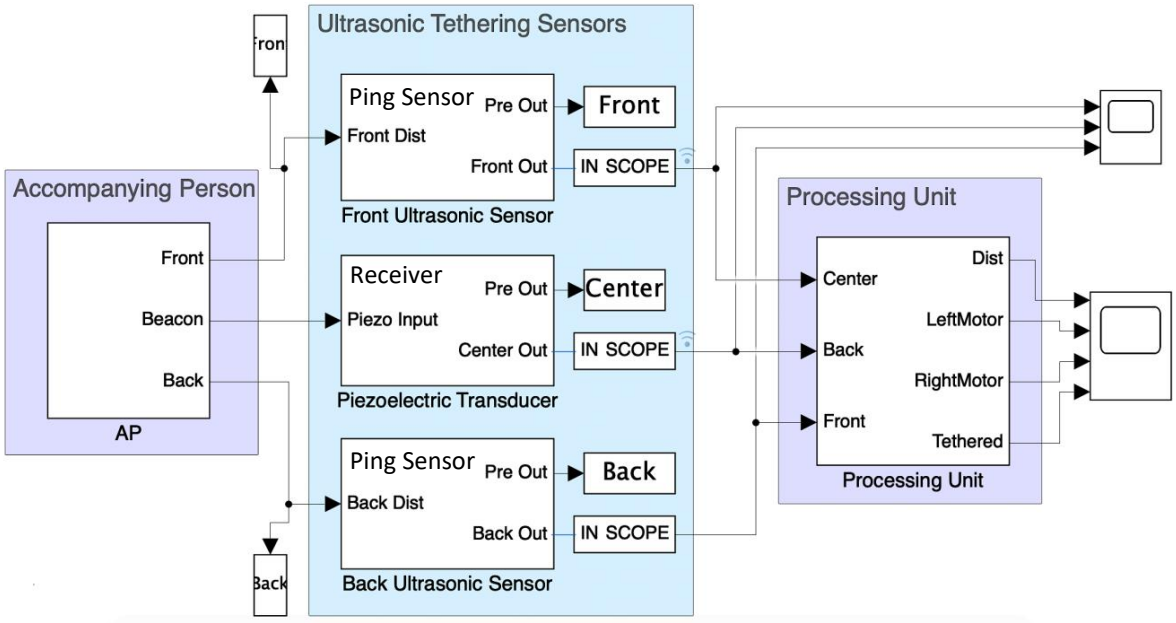
3.4 Model design and development

The simulation model consisted of three main sub-systems: AP module, ultrasonic receiver module, and processing unit. Identifying and determining the AP pose was simulated by using the

AP sub-system and the ultrasonic tethering sensor sub-system. Wheelchair navigation was achieved by determining wheelchair motor direction signals in the processing unit. Ultrasonic sensor placement for triangulation is shown in Figure 3.1 (a) . Two ping sensors were used to determine the pose and one piezoelectric receiver and a transmitter (beacon) on the AP were used for identification (Figure 3.1 (b)). Identification is the process of successfully identifying the accompanying person or the process for unsuccessful identifications.



(a)



(b)

Figure 3.1 (a) Sensor placement on the wheelchair and beacon [53], and (b) Ultrasonic tethering model in Simulink. Ping sensor transmits and receives ultrasonic pulses. Piezoelectric receiver receives ultrasonic pulses.

3.4.1 Accompanying person sub-system

The AP sub-system (Figure 3.3) consisted of a beacon that transmitted a 42 KHz ultrasonic signal for identification. This beacon would be worn on the AP side and consisted of an ultrasonic transmitter module and three message signals that modelled AP walking, turning, and stopping.

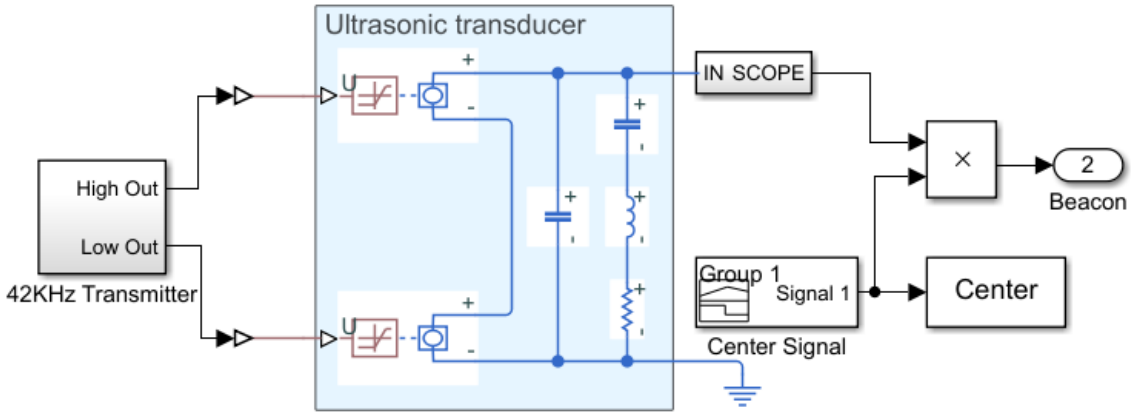


Figure 3.2 Accompanying person sub-system for model identification, walking, turning, and stopping.

The ultrasonic signal was produced using a pulse generator and a Simulink generic linear sensor with a simplified Van Dyke equivalent RLC circuit model for a piezoelectric transducer. The circuit components and parameters for the Van Dyke model are real numbers; however, other models (Sherrit model, etc.) are complex numbers [55]. Real numbers are useful for simulations using real-world parameters. The message signals were used as the sensor's voltage, proportional to the AP distance.

These signals were modulated with the 42 KHz signal to simulate the AP pose. The front, back and center signals used for simulation, produced by the Simulink generic linear are shown in Figure 3.3. The signals were modeled using the signal builder module in Simulink and were designed using equation 3.1 and 3.2.

$$a = m * \theta \quad \textbf{(3.1)}$$

$$b = n * \theta \quad \textbf{(3.2)}$$

where, a and b are the signal amplitudes, m and n are constants of proportionality, θ is the AP orientation angle and d is the AP distance from the sensor. The 42 KHz pulse generator was designed to produce 0 – 5 V square wave pulses with a 50% duty cycle using a PWM generator.

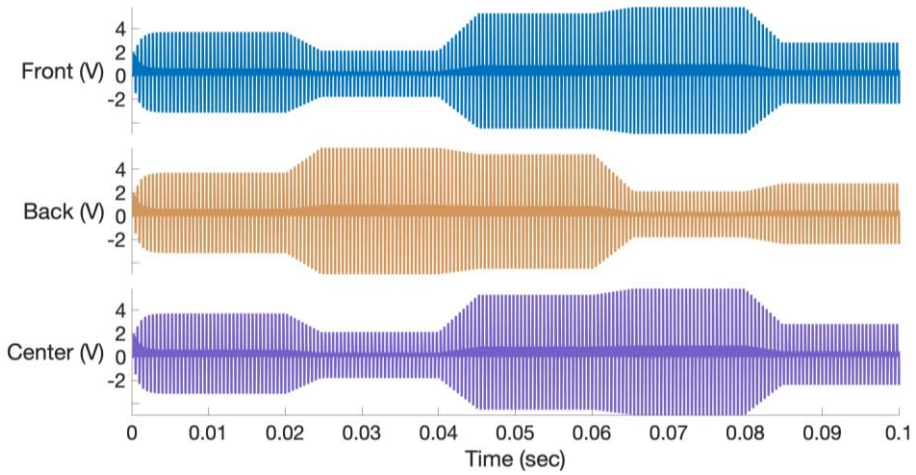


Figure 3.3 Ultrasonic signals produced by the front, back and center generic linear sensors. Signals are amplitude modulated square wave pulses of 42 KHz with 50% duty cycle.

The transmitter circuit consisted of a voltage doubler, a voltage inverter and an oscillator as shown in Figure 3.4. The voltage doubler that converted digital pulses from 0 – 5V to 0 – 20V and a voltage inverter that inverted and pulled down the 0 – 20V to ±10V to drive a piezoelectric transducer. The oscillator was used to produce pulses for doubling the voltage using switching capacitors C1, C2, C3 and C4.

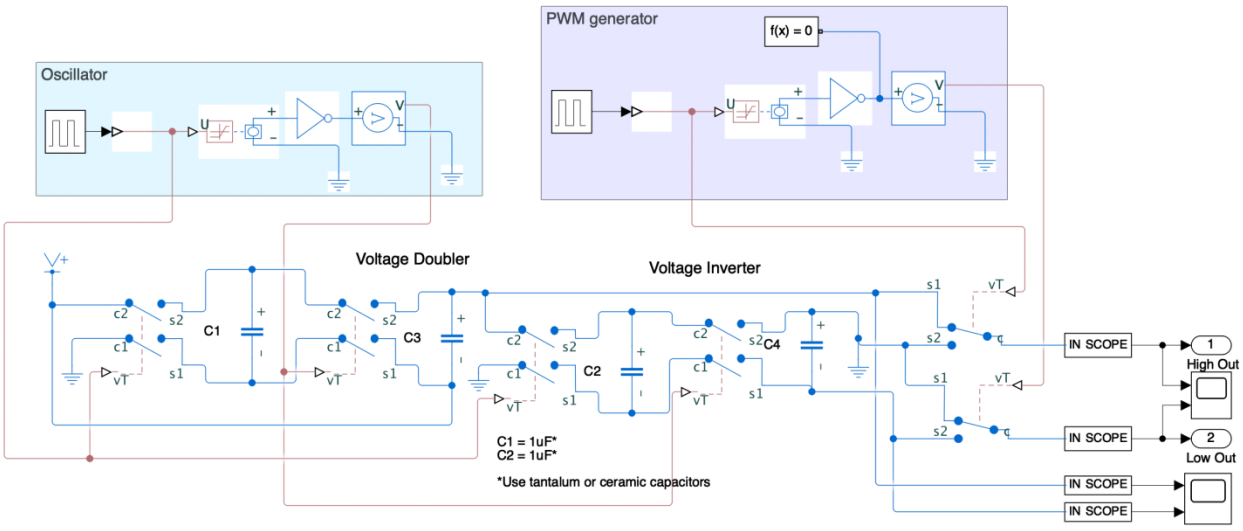


Figure 3.4 The 40 KHz 10 V p-p pulse generator module that consists of a voltage doubler, a voltage inverter, an oscillator and a PWM generator.

3.4.2 Ultrasonic sensor module

The ping sensor module was designed to determine the AP pose by producing an analog voltage proportional to the AP distance. The analog voltage output is given by equation 3.3 [38]:

$$D = \frac{v_{sound} * ToF}{2} \quad \textbf{(3.3)}$$

where *D* is the AP distance (tether distance), v_{sound} is the speed of sound in air and ToF is the time for the pulse to travel from the sensor to the AP and the time for the reflection from the AP to the sensor (round trip time-of-flight). The system consisted of two identical range sensor modules to measure the distance from two distances for triangulation. The sensor module consisted of an ultrasonic pulse generator and amplification and filtering circuits. The ping sensor consisted of two Simulink generic linear sensors, to convert the pulsed output from the transmitter and modeled as the piezoelectric transducers present in an ping sensor (Figure 3.5). Each of the generic linear sensors had an input to output signal ratio of 1:100 to model output signal behavior and attenuation for real-world situations.

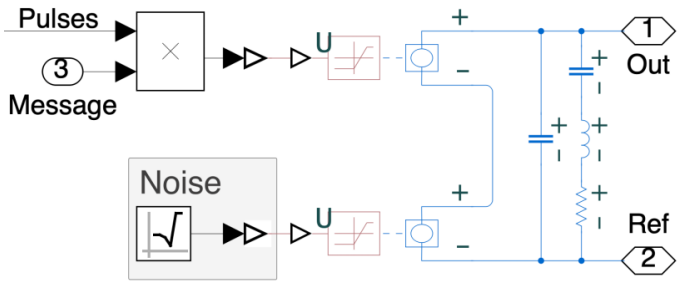
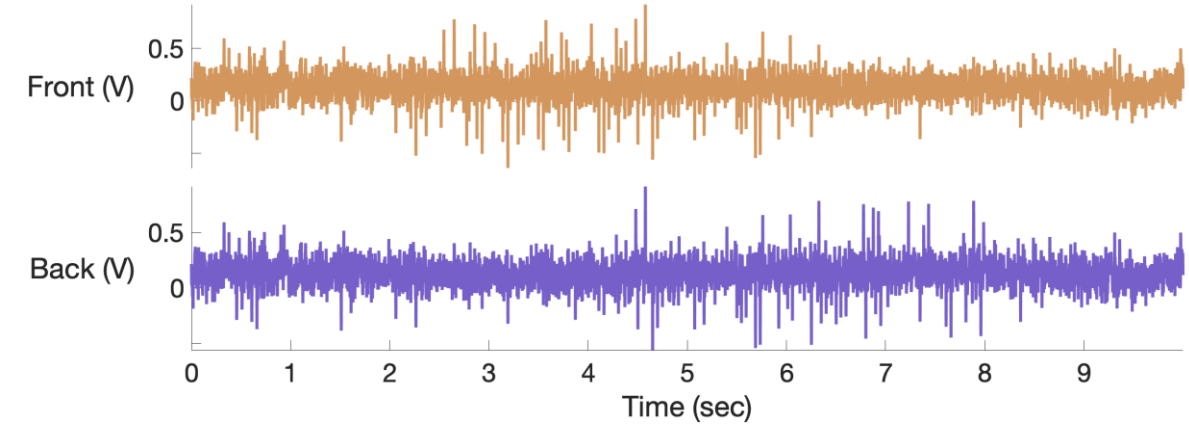
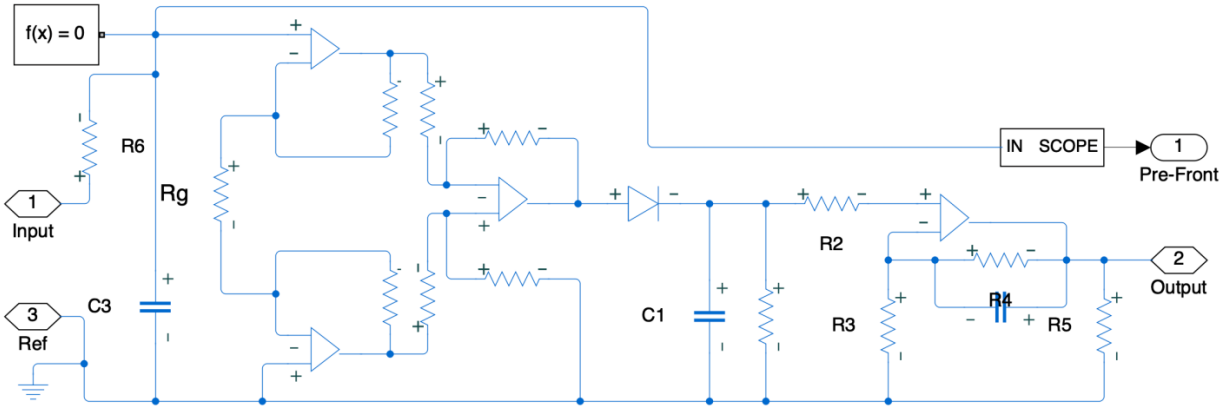


Figure 3.5 The ping sensor module for determining pose that consists of a multiplexer, two Simulink generic linear sensors (modelled as ping sensors).

Noise was modulated to the message signal (Figure 3.6 (a)) and was amplified and filtered using the amplification and filtering circuits simulated using the Simscape toolbox on Simulink [56]. The amplification circuits (Figure 3.6 (b)) amplified the signal change from a millivolt range to a microcontroller readable 0 – 5V range, using an amplifier. The filtering circuit included an active low pass filter and diode-based amplitude demodulator to extract the message signal from amplitude modulated ultrasonic signal. This determined the voltage proportional to AP distance.



(a)



(b)

Figure 3.6 (a) Noisy signal generated by front and back ping sensors and, (b) Ultrasonic sensor amplification and filtering circuit that consists of an instrumentation amplifier and low-pass filter circuits.

The resistor and capacitor values (Rg, R1, C1 and C3) for each sensor circuit was determined and optimized using Simulink Design Optimization toolbox, specifically, the Parameter Estimation tool. The criteria used for estimating the circuit parameters were:

- High signal-to-noise ratio.
- Output signal limit between 0 and 5V to eliminate information loss due to ADC clipping.
- Passive components with values that are readily and commercially available.

3.4.3 Piezoelectric ultrasonic receiver module

The piezoelectric ultrasonic receiver identified the AP within the receiver's field of view by reading the beacon signal (Figure 3.7) [53]. The receiver included an ultrasonic transducer and amplification and filtering circuits (Figure 3.6). A noise function was added to the beacon message signal and was amplified and filtered using an identical amplification and filtering circuits from Section 3.4.2. Figure 3.7 shows the noisy signal before amplification and filtering.

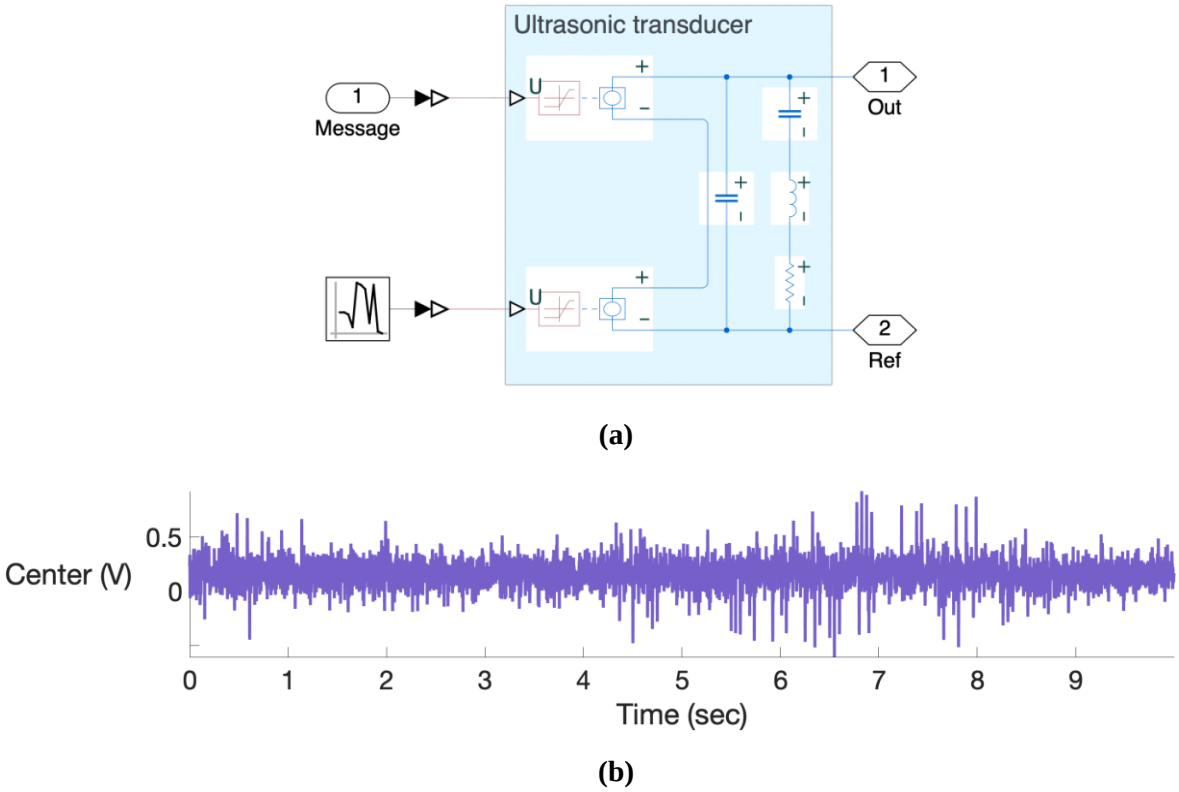


Figure 3.7 (a) Piezoelectric ultrasonic receiver module in Simulink, and (b) noizy signal produced by the piezoelectric ultrasonic receiver.

3.4.4 Processing unit

The processing unit simulated a microcontroller and was used to process the sensor signals and produce motor drive control signals using a Matlab function block (shown in Figure 3.8). The Matlab function block determined left and right motor direction signals by calculating the tether distance from the front and back, and identified the AP from the center sensor signals using thresholds and comparing the front and back sensor signals. Comparing front and back signals indicated the forward and backward AP direction. The tether distance is given by equation 3.4,

$$D = \sqrt{\frac{L^2 + T^2}{2M}} \quad \textbf{(3.4)}$$

where, L and T are distances from two ping sensors to the AP and M is the distance between the two sensors from Figure 3.1. Optimized thresholds were based on the expected AP motion (i.e., for every AP direction, the expected wheelchair direction was calculated from the tether distance, using equation 3.4).

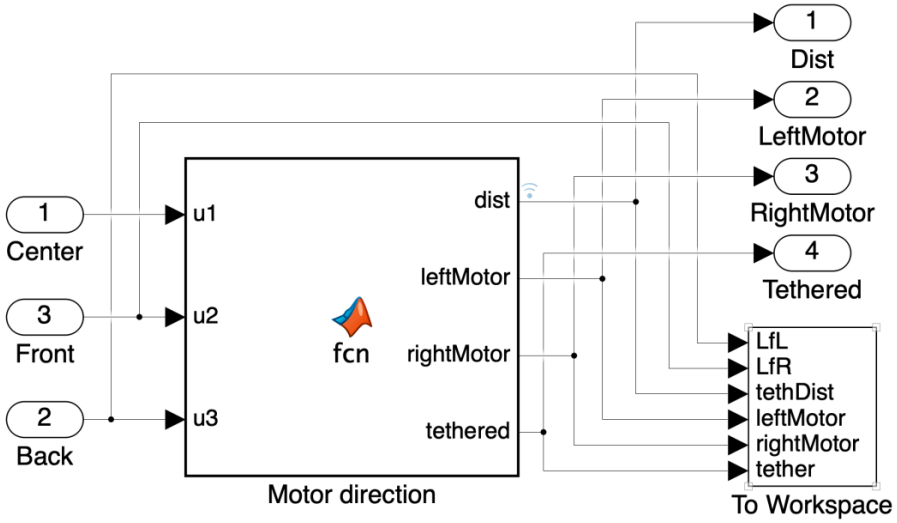


Figure 3.8 The Matlab function block that simulated a processing unit.

3.5 Results

The system was simulated using an ode15s solver on Simulink and all outputs were stored as Matlab variables for displaying using the To Workspace block. An average of 7 iterations occurred for optimizing each parameter for different conditions; such as, AP walking, turning, and stopping. A sample output from the Parameter Estimation tool is shown in Figure 3.9. The sample output shown in Figure 3.9 is for the gain resistor (Rg), R2 and R3 for the three amplifiers, one for the piezoelectric receiver and two for the ping sensors. The circuits had an average signal-to-noise ratio of 7.65 dB, a gain of 495 and cut-off frequency of 52Hz.

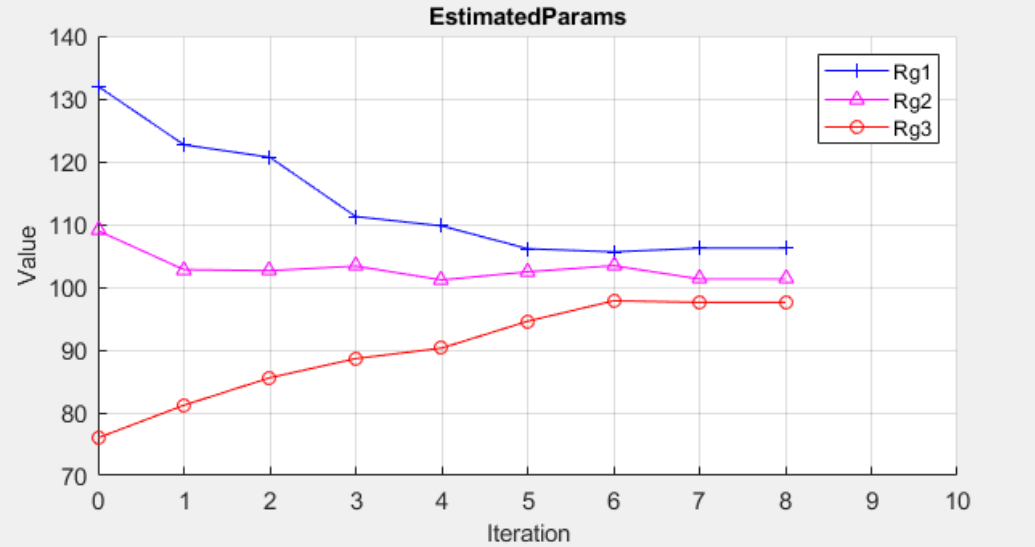


Figure 3.9 Sample output from the Parameter Estimation tool to optimize the gain resistor Rg used for different conditions. The parameters converge at around 100 Ohm at the seventh iteration.

Table 3-1 shows the average optimized and final value parameters for each component for three circuits used in the simulation.

Table 3-1: Optimized component parameters using the Parameter Estimation tool and the final chosen values for each component.

Component	Optimized Value	Final Value
Rg	101.7 (Ohm)	100 (Ohm)
R1	9.2 (KOhm)	10 (KOhm)
R2	1.1 (KOhm)	1.3 (KOhm)
R6	1.4 (KOhm)	1.6 (KOhm)
C1	20.23 (nF)	22 (nF)
C2	0.1 (uF)	0.1 (uF)
C3	0.12 (uF)	0.1 (uF)

Optimized thresholds for the calculated tether distance are shown in Table 2. Sample results for the filtered front and back signals, and the AP distance from the sensors are shown in Figure 3.11 (a) [53].

Table 3-2: Optimized thresholds calculated from the average tether distance for each AP direction.

AP direction	Avg. tether distance (Volts)	Thresholds (Volts)
Forward/Back	1.9	± 0.4
Right	1.4	± 0.4
Left	0.4	± 0.4
Stop	1	± 0.4

Sample wheelchair motor direction results for the left and right motors are shown in Figure 3.10. Wheelchair front and back direction was calculated by comparing the signals from the front and back sensors with each other. The wheelchair direction can be determined by using the left and right motor signals in a differential drive system.

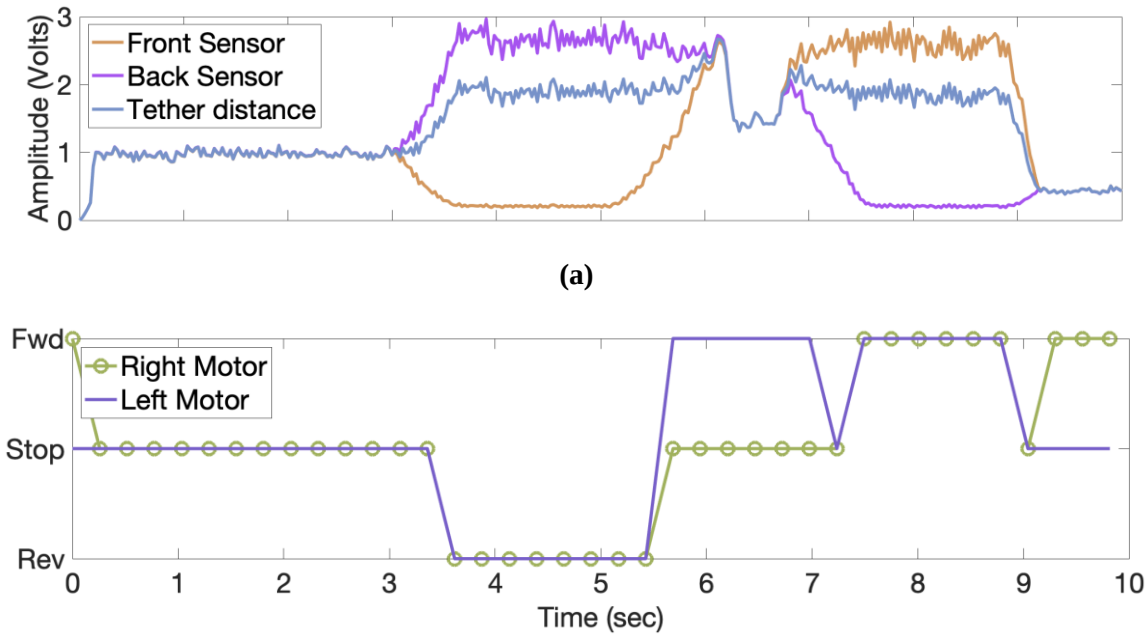


Figure 3.10 (a) Post-processed front (orange) and back (magenta) ping sensor signals to calculate tether distance and calculated tether distance (blue), and (b) Direction of left and right wheelchair motors.

3.6 Conclusions

Ultrasonic tethering can be used for automatic powered wheelchair navigation beside a person accompanying. This paper simulated a smart wheelchair system with non-contact tethering to an accompanying person by determining the person's position with respect to the wheelchair. The proposed ultrasonic tethering system was modelled and simulated in Matlab® and Simulink®. The amplification and filtering circuit parameters were optimized to achieve required gains and cut-off frequencies for the circuits. The results showed that the accompanying person's position can be calculated using two ultrasonic sensors from which wheelchair motor signals can be produced. The simulation test criteria included testing different scenarios that included identifying and determining the person's forward, reverse, left and right motion with respect to the wheelchair. The system was designed to stop when the AP is outside the sensor field of view since the AP is considered untethered at that point. The simulated electrical circuits determined the optimum amplification and filtering circuits with active and passive components required to build an ultrasonic tethering prototype.

4 Ultrasonic Tethering to Enable Side-by-Side Following for Powered Wheelchairs

4.1 Overview

This chapter discusses the ultrasonic tethering system hardware design and development and aids in addressing objective 1. All electronic circuit designs for the prototype can be found in Appendix A. All designs related to 3-D printing the prototype can be found in Appendix B. This chapter consists of a modified journal manuscript:

> Pingali T, Lemaire ED, Baddour N, Ultrasonic Tethering to Enable Side-by-Side Following for Powered Wheelchairs, Sensors, 19(1), 109, 2018.

4.2 Abstract

In social situations, people who use a powered wheelchair must divide their attention between navigating the chair and conversing with people. These conversations could lead to increased mental stress when navigating and distraction from maneuvering the chair. As a solution that maintains a good conversation distance between the wheelchair and the accompanying person (Social Following), a wheelchair control system was developed to provide automated side-by-side following by wirelessly connecting the wheelchair to the person. Two ping sensors and three piezoelectric ultrasonic transducers were used to identify the accompanying person and determine their position and heading. Identification involved an ultrasonic beacon worn on the person's side, at hip level, and receivers on the wheelchair. A plug-and-play prototype was developed and connected to a Permobil F3 Corpus wheelchair with a modified Eightfold Technologies SmartChair Remote. Results demonstrated that the system can navigate a wheelchair based on the accompanying person's trajectory, which is advantageous for users who require hands-free wheelchair control during social activities.

4.3 Introduction

Traditional joystick-controlled powered wheelchairs require users to understand their surroundings, perceive space (depth/color), and physically control the joystick [1]. Due to reduced vision, cognitive deficits, or motor-neuron diseases, a segment of users have difficulty using

joysticks, which could lead to increased dependency on caregivers, mental stress, depression, social anxiety, and isolation [5], [57]. Thirty-one percent of persons with mobility disorders are frequently depressed due to these factors [5]. Another issue is 'distracted navigation', which could lead to accidents such as tipping/falling and bumping into curbs, trees, or persons [2], [19].

Recent advances in wheelchair technology have enabled smart wheelchairs, an extension of powered wheelchairs that use an embedded computer and sensor systems to assist navigation [30]. This intelligent assistive device incorporates technology from autonomous mobile robots and requires minimum user involvement for navigation. A typical smart wheelchair control has an input method, a processing device, and a drive controller. Input methods include joystick or gesture-based control using head posture, eye-gesture, voice commands [25]–[28]. Smart wheelchair assisted navigation can also include object following, also termed tethering. In this research, tethering is defined as the process of human-following to assist in powered wheelchair navigation.

Tethering techniques can be grouped into wired/contact tethering or contactless tethering. Contact tethering is loosely based on dog-on-a-leash, with a mechanical tether connecting the AP to the wheelchair. For example, Chu et al. [43] used a mechanical string tether between a mobile robot and a person and measured the string tension and angle to calculate the person's trajectory and determine the mobile robot's direction and velocity. Na et al. [46] used a rod and reel mechanism to calculate a mobile robot's speed for following. Contactless tethering would be advantageous over wired/contact tethering since wired/contact-based tethering could hinder free movement and maneuvering around obstacles between the AP and wheelchair. Contactless tethering use sensors like Lidar, cameras, or range-sensors to determine an object's position and feed a control system to maneuver the wheelchair [43]. Examples include a human following mobile robot using laser range scanners to determine the shin position of a person in front, and then match the robot to the person's trajectory [44] and a caregiver following wheelchair using omni-directional vision that wirelessly tethered the wheelchair to a caregiver [47].

Reviewing previous human-accompanying research [29], [47]–[49], [51], it can be concluded that tethering involves identifying the AP and determining the AP's pose, which can be achieved using commercially available infrared range, ultrasonic range, cameras, or Lidar. For example, Kobayashi et al. [47] developed a caregiver following wheelchair that determines caregiver position using laser range sensors and extracted the contour of the shoulders as a means to track the caregiver movement, and an-omni-directional camera to identify the caregiver. The human

following smart wheelchair [48] uses a LRS and an ultrasonic sensor ring to identify and determine the person's pose respectfully. The ultrasonic sensor ring is used to determine the distance of the person in front of the wheelchair using triangulation. Ultrasonic sensors produce a high-frequency sound pulse to determine the distance of an object in its field-of-view. The distance is measured using Time-of-Flight (ToF) [36, p. 139]. Unlike light-based sensors (e.g., cameras, Lidar, infrared range sensors), ultrasonic sensors are not affected by the color or transparency/texture of the person's clothes, can be used in low-light or no-light situations (e.g., in the dark or at night), are not affected by dust, smoke and mist [58], and have a resolution of 25 mm and a range from 100 mm to 6451 mm [59]. Lidar and LRS require a computer to process sensor information, making the system more expensive and bulkier.

This chapter proposes a plug and play powered wheelchair control system to wirelessly tether the wheelchair to follow alongside an AP. This will allow the wheelchair user to converse with the AP without needing to physically control the wheelchair, thereby achieving the desired objective of social following, reducing mental stresses, and enhancing safety when maneuvering and navigating the wheelchair. This research implemented automated side-by-side following using ultrasonic tethering, with the SmartChair Remote [23] enabling seamless wheelchair control from the UT system to the joystick, and vice versa. The main objective of this research was to develop a system that would initiate a casual proxemic communication distance [52] between the AP and the powered wheelchair, and test the ultrasonic tethering system for contactless tethering between a powered wheelchair and an AP.

4.4 Methodology

The design criteria for the social following powered wheelchair system were:

- Maintain conversation distance between 60 and 180 centimeters [52]
- Maintain tether when a person walks alongside the wheelchair and break tether when the person is no longer available
- Work with the powered wheelchair joystick controller, to allow user to retake joystick control at any time
- Perform as intended in low light, inexpensive and easy to connect and attach/detach to any powered wheelchair.

An ultrasonic tethering approach was selected to meet these design criteria. The prototype of the proposed system mounted on a Permobil powered wheelchair is shown in Figure 4.1.

(a) **(b)** **(c)**

Figure 4.1 (a) Ultrasonic tethering system prototype mounted on a Per4mobil F3 corpus powered wheelchair, (b) and (c) Ultrasonic tethering system for social following.

4.4.1 System architecture

The system architecture consisted of two main processes to achieve ultrasonic tethering: identify the AP and determine the AP's pose. Identification and pose estimation were achieved using commercially available ultrasonic sensors and a microcontroller. Ultrasonic sensors were chosen due to their small form-factor, ability to accurately detect objects within a short distance in different environments, and low processing power requirements for signal analysis.

4.4.2 Identifying the accompanying person

In social situations, wheelchair navigation beside a person requires consistently identifying the person to tether and accompany, and avoiding wheelchair navigational error from intermittent tethering to by-standers or other objects beside the wheelchair. Three piezoelectric ultrasonic transducers on the wheelchair's side and a beacon on the AP were used to identify the AP in the transducer field of view (detection area beside the wheelchair), as shown in Figure 4.2. AP beacon ultrasonic signals were received by piezoelectric ultrasonic receivers on the wheelchair's side. The receivers produced an analog signal when the person was in the field of view.

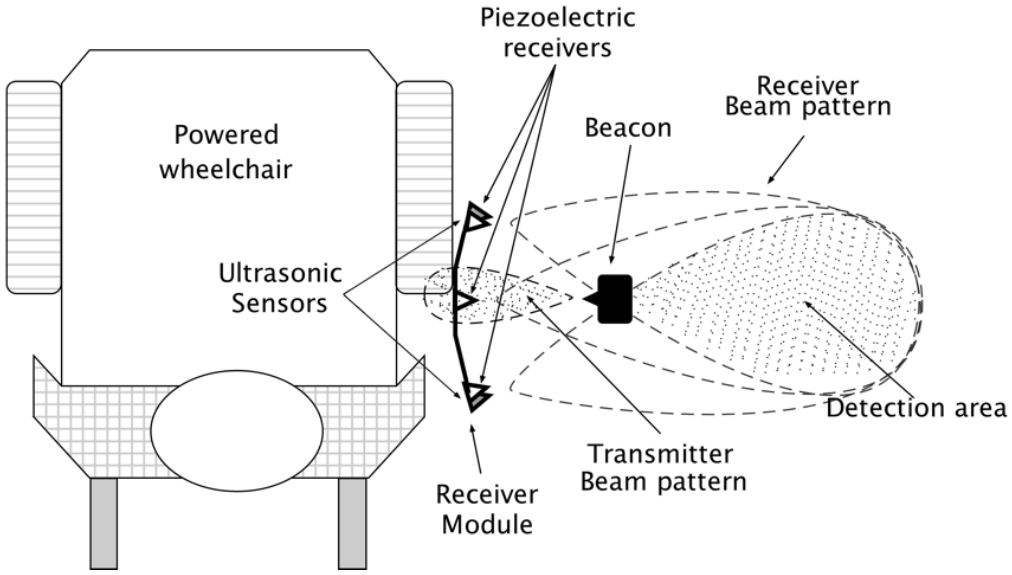


Figure 4.2 Detection area beside the powered wheelchair for AP identification. Beam patterns converge at the center of where the person is expected to walk.

4.4.3 Determining accompanying person position and orientation

Maintaining an appropriate conversation distance between the AP and wheelchair requires the relative AP position and heading. AP position beside the wheelchair was calculated by determining the distance between the wheelchair center and the AP, and calculating the AP angle with respect to the wheelchair. This position can be represented using a coordinate system with the receiver module as the origin (Figure 4.3).

Position and heading are calculated using triangulation [60]–[62]. Two ping sensors placed at the wheelchair side (facing the AP) were used to determine the distance from the sensors to the AP side (tether distance). The tether distance (D) is expressed as the total distance (conversation distance (d_c)) subtracted from the sum of the distance from wheelchair's sagittal plane to the receiver module (d_1) and the distance from AP sagittal plane to beacon (d_2). As illustrated in Figure 4.4, a tether distance smaller than the conversation distance can be calculated using equation 4.1.

$$D = d_c - (d_1 + d_2) \quad \textbf{(4.1)}$$

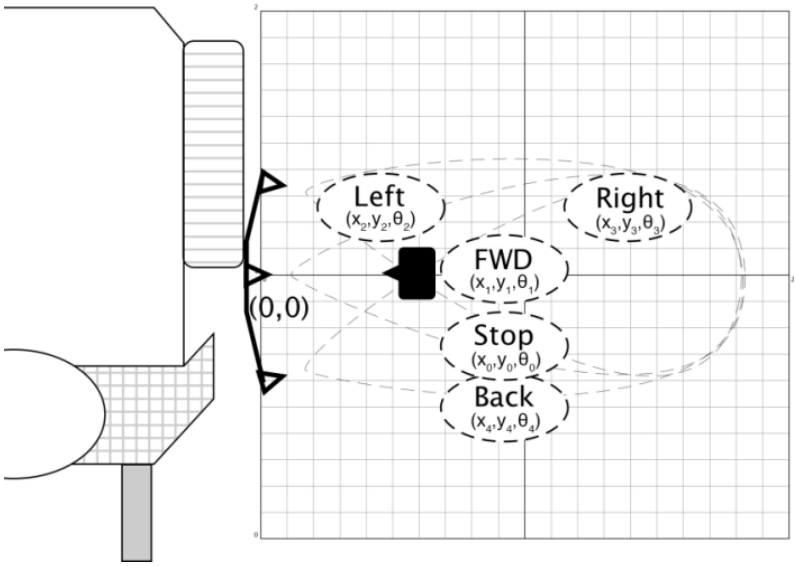


Figure 4.3 AP position in coordinate space beside the wheelchair. x and y represent location coordinates and θ is the angle of a line from the origin to the point x, y.

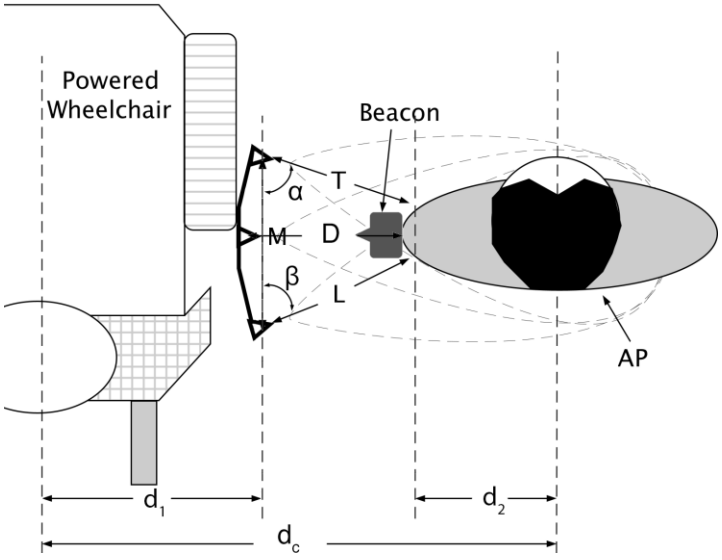


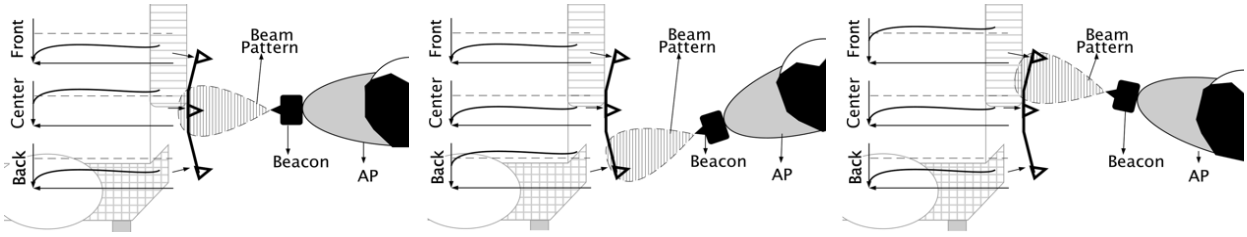
Figure 4.4 Relationship between the AP conversation distance (d_c) and tether distance (D). T and L are distances between the Ping sensors and AP. α and β are angles between Ping sensors and AP. M is distance between Ping sensors. d1 is distance from wheelchair sagittal plane to the receiver module. d2 is distance from AP sagittal plane to beacon.

Ping sensors produce an analog voltage proportional to the distance of an object closest to the sensor. This distance was calculated from the sound pulse ToF, by measuring the time for the pulse to travel from the sensor to the object and the time for the reflection from the object to the sensor [63]. The distance from the sensor to the person was calculated using equation 4.2 [38]:

$$L,T = \frac{\left(Velocity\ of\ sound\ in\ air * ToF_{(L,T)}\right)}{2} \quad (4.2)$$

4.4.4 Identification and pose detection algorithm

AP identification and direction of motion were determined by comparing signals from the ping sensors and piezoelectric ultrasonic transducers to thresholds. AP identification during walking, stopping, and turning occurred by comparing the piezoelectric ultrasonic receiver signal to a threshold (Figure 4.5). While walking, post-processed signals from the ultrasonic transducers are modulated as sine waves that are out of phase with each other, due to beacon angular motion at the hip (i.e., the beacon faces one transducer at a time, causing signal attenuation of the other transducers (Figure 4.5)).



(a) **(b)** **(c)**

Figure 4.5 Ultrasonic transducer analog output while the person is (a) walking and stopping, (b) turning right, (c) turning left. Dashed line is high threshold.

Figure 4.6 shows thresholds used by the receiver to identify the AP and determining if in-motion or static for three trials. The AP is identified when the center piezoelectric receiver signal is greater than the lower threshold. When walking straight, the center signal is greater than the high threshold. The AP stop and turning condition is when the center signal is between the low and high thresholds. The center signal amplitude is proportional to the angular motion at the hip.

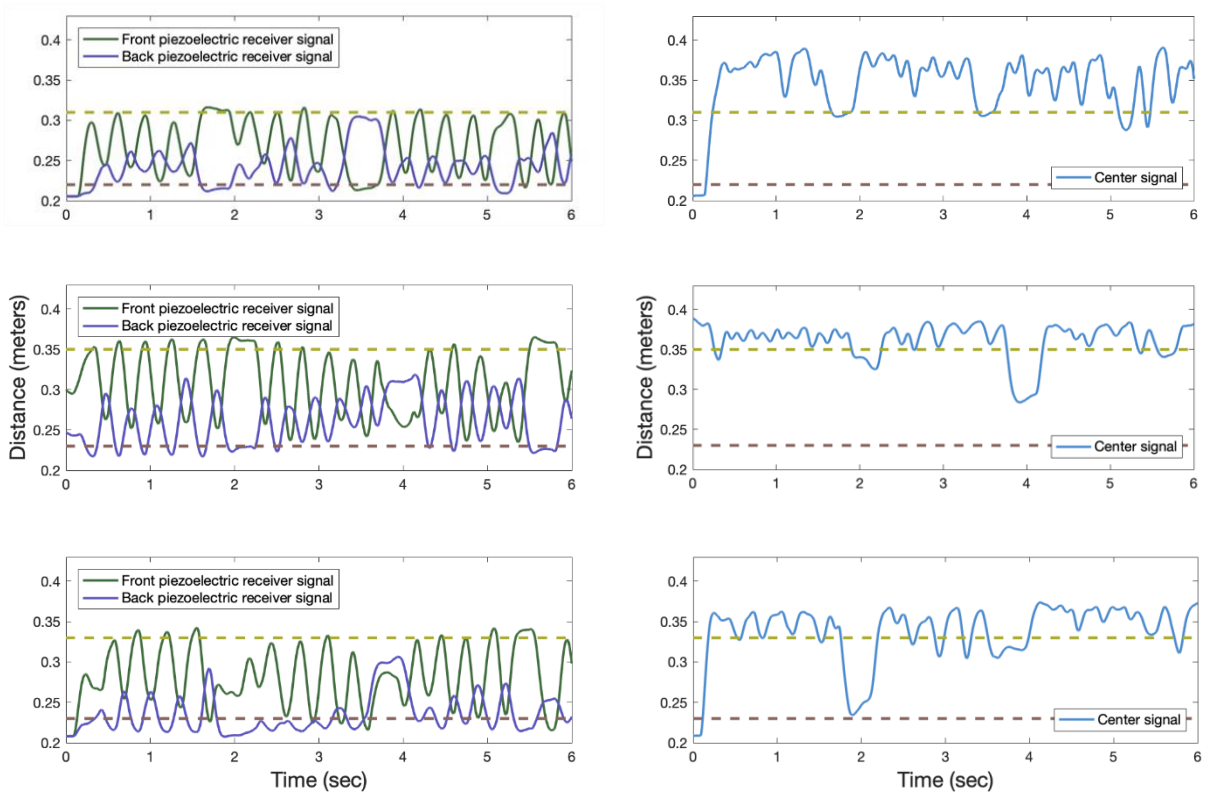


Figure 4.6 (a) Signals from front and back receivers for three trials. (b) Thresholding used on the center receiver signal to determine if the person is in-motion or static for three trials. Dashed lines represent the thresholds.

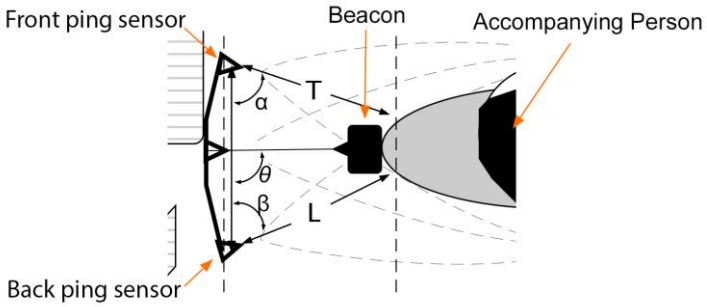


Figure 4.7 Nomenclature used for the algorithm. M is the distance between the front and back sensors. T and L are distances from the sensors to the AP side. α and β are angles formed between M and T or L and θ is the angle formed between M and the tether distance.

To determine AP position and heading, two ping sensors were placed at a known distance (M) on the wheelchair's side (Figure 4.7). Two distances to the AP (L, T) were determined from the sensor output. A line from the sensor center to the closest point on the AP was used to calculate two angles (α, β) using the Law of Cosines, equation 4.3 and 4.4,

$$\alpha = \arccos\left(\frac{T^2 + M^2 - L^2}{2TM}\right) \quad \textbf{(4.3)}$$

$$\beta = \arccos\left(\frac{L^2 + M^2 - T^2}{2LM}\right) \quad \textbf{(4.4)}$$

The angle of AP with respect to the receiver center module can be calculated using equation 4.5:

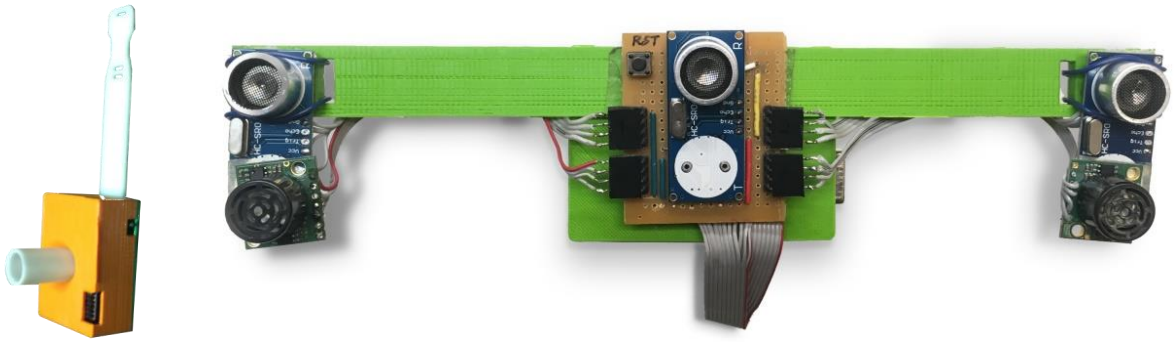
$$\theta = 180 - \left[\left(\frac{180 - (\alpha + \beta)}{2}\right) + \alpha\right] \quad \textbf{(4.5)}$$

The distance between the closest part of the AP that reflects the ultrasonic sensor echoes and the center of the receiver module, can be calculated using equation 4.6:

$$\text{Tether distance} = \text{sqrt}\left(\frac{L^2 + T^2}{2M}\right) \quad \textbf{(4.6)}$$

4.5 Prototype

A prototype was developed using cost-effective 16 mm piezoelectric ultrasonic transducers, MaxBotix MB1010 ping sensors, ATmega 328 microcontroller, and components such as operational amplifiers, resistors, capacitors, diodes, and 3-D printed cover (Figure 4.8).



(a) **(b)**

Figure 4.8 (a) Ultrasonic beacon for AP, (b) ultrasonic tethering receiver for powered wheelchair (not to scale).

4.5.1 Accompanying person beacon

The AP beacon transmits ultrasonic signal pulses toward the wheelchair (Figure 4.8 (a)). This beacon hooks on the AP at the waist (belt, pocket, etc.), facing towards the wheelchair, and produces 40 KHz signal pulses using a piezoelectric ultrasonic transmitter and Atmel 328 based microcontroller as a pulse generator and voltage driver circuit. The microcontroller generates two 40 KHz pulses with 180-degree phase shift, using two built-in 16-bit timers. To achieve the required voltage to drive the 16 mm piezoelectric ultrasonic transducer, a voltage driver circuit based on a TTL to TIA232 converter was modified to generate 20 V from a 5 V TTL logic. The beacon housing was 3-D printed using Polylactic Acid (PLA) and houses a 9 V battery.

4.5.2 Receiver on the powered wheelchair

A receiver module with three piezoelectric ultrasonic receivers (spaced 12 cm apart), two MaxBotix ping sensors (spaced 24 cm apart), signal amplification and filtering circuits, microcontroller, and 3-D printed box was fixed to the side of the wheelchair (Figure 4.8 (b)). The MaxBotix range sensors were placed under the piezoelectric receivers. This module identified and detected AP position and heading in the ultrasonic transducer field of view. AP identification was achieved using three piezoelectric ultrasonic receivers placed in a concave pattern on the powered wheelchair's AP side. AP position and heading were determined using two MaxBotix range sensors, oriented inward by 15-degrees. Range sensor placement was designed to point the sensors toward the center of a 30 cm radius circle, for optimal tether distance. These inexpensive ultrasonic sensors provided ranging from 0 to 254-inches, with 1-inch resolution with a narrow beam pattern [59]. All analog signals are converted to digital signals using the analog to digital converter inside the Atmel 328 based microcontroller. The microcontroller was also used to determine the wheelchair direction based on AP position. Figure 4.9 shows the receiver working block diagram.

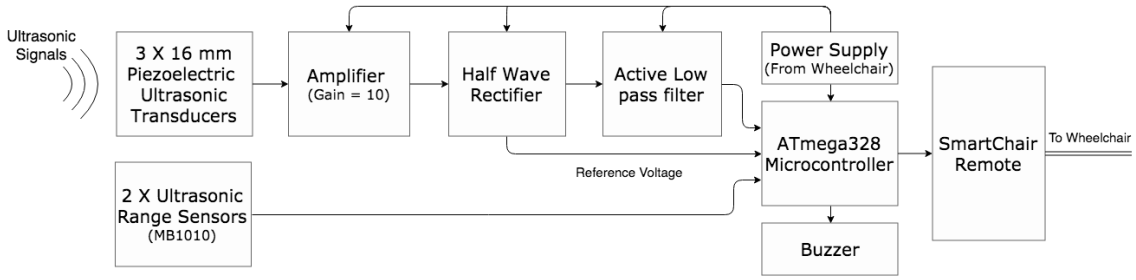
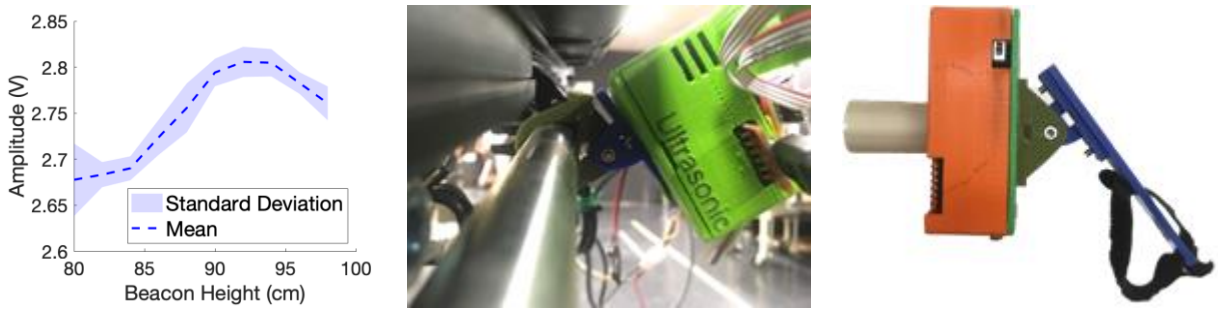


Figure 4.9 Receiver module hardware block diagram

4.5.3 Effect of accompanying person's height on the tethering system

Initial prototype development included a fixed beacon on the AP and receiver on the wheelchair. The beacon was attached to the AP with a flat belt hook that pointed the beacon above the receiver; therefore, the beacon was out of the receiver's line-of-sight. Especially for shorter or taller people, the lack of a line-of-sight connection resulted in the ultrasonic tethering system unidentifying the AP while walking. To study the effect of AP waist height on the tethering system, a test was conducted by mounting the beacon on a tripod and changing the beacon height while measuring the piezoelectric ultrasonic receiver input signal on a receiver module that was mounted on the wheelchair. The tripod height was varied from 80 cm to 100 cm, in 2 cm intervals. Experiment results are shown in Figure 4.10 (a). The output signal changed by 10 mV, on average, for every 2 cm change in height.

To counter the beacon height effect, a new hinge design was added to provide varying beacon angles on the AP and the wheelchair receiver mount, thereby making it easier for the AP and the wheelchair user to manually rotate the beacon or receiver to the same line-of-sight (Figure 4.10).



(a) **(b)** **(c)**

Figure 4.10 (a) The piezoelectric transducer mean and standard deviation signal output for beacon heights ranging between 80 cm to 98 cm, (b) the ultrasonic tethering receiver mounted to the wheelchair using a movable hinge, and (c) the beacon with a movable hinge.

4.5.4 Accompanying person feedback and ultrasonic tethering system integration with the powered wheelchair

The receiver module was mounted, using 3D-printed U-clamps and movable hinges, on a Permobil F3 corpus powered wheelchair. A buzzer was included in the receiver module to produce a different tone for each change of mode. In situations where the tether is lost, the buzzer would sound an alarm for a few seconds, notifying the AP to move into the sensor field of view. In situations where deliberate loss of tether occurs, the buzzer alarm would be ignored, and the system would automatically switch to joystick control. The onboard Atmel328 based microcontroller transmits wheelchair direction commands as characters to a modified Eightfold Technologies SmartChair Remote [23] via serial communication. The system outputs characters based on the thresholding algorithm output: front (f), back (b), right (r), left (l), stop (s). The SmartChair Remote emulates a joystick and was connected in parallel to the wheelchair mechanical joystick, producing similar voltages as the joystick. SmartChair Remote modifications included a read data function from the ultrasonic tethering system.

Table 4-1 Ultrasonic tethering system modes of operation.

AP	System Operation	Wheelchair Control/Motion	Buzzer Operation
Unavailable, not tethered	Not tethered	Joystick Control	No output
Calibration	Sensor Calibration	Stop	Multitone beep at calibration start and stop
Available, Tethered	Tethered	Stop	No output
Available, Tethered	Tethered	Forward	No output
Available, Tethered	Tethered	Backward	No output
Available, Tethered	Tethered	Right	No output
Available, Tethered	Tethered	Left	No output
Loss of tether during operation	Not tethered	Joystick Control	Continuous single tone beep

4.5.5 Sensor calibration

The ultrasonic tethering system calculates thresholds based on the ultrasonic receiver and ping sensor information. For calibration, the AP stands next to the wheelchair, in the sensor field-of-view, for five seconds, at a comfortable distance for conversations between the AP and the wheelchair user. The system reads all sensor data and calculates low and high thresholds for tether distance, tether angles and identification thresholds based on equation 4.7.

$$\begin{bmatrix}\text{threshold}_L\\ \text{threshold}_T\end{bmatrix} = x \pm \begin{bmatrix}\frac{1}{n} * \sum_{i=0}^{n} K_{d1} * L_i + K_{d2} * L_{i-1}\\ \frac{1}{n} * \sum_{i=0}^{n} K_{d1} * T_i + K_{d2} * T_{i-1}\end{bmatrix} \quad \textbf{(4.7)}$$

where, K_{d1} and K_{d2} are filter gains (0.8987 and 0.1013, respectively), L and T are the front and back sensor signals, n is the number of iterations, and x is the low or high gain. The thresholds for the tether distance were calculated using equation 4.6 and the threshold for the tether angle was calculated using equation 4.5. The effect of AP height on the receiver further concluded that the change in amplitude could also exist while walking because of lateral hip movement, which could result in unidentifying the AP while walking. To fix the effect, a factor of 0.0735 was added to the sensor calibration function for calibrating thresholds.

4.6 Ultrasonic tethering system experimental test protocol

System performance was evaluated using sensor data acquired during three trials, where the AP and the wheelchair followed a fixed straight path for six seconds while data were acquired at 166 Hz. Two parallel lines were taped on the laboratory floor at 90 cm apart, the distance between the wheelchair center and person center (dc), as shown in Figure 4.11. The distance from wheelchair sagittal plane to the receiver module center (d1) was 40 cm and the distance from the AP sagittal plane to beacon (d2) was approximately 20 cm. The distance from the AP sagittal plane to beacon (d_2) was measured using a tape measure, from the sagittal plane to the beacon's face (in practice, this distance would be calculated by the system during calibration). From equation 4.1, the tether distance was 30 cm. Wheelchair drive control was disabled to allow the user to drive the powered wheelchair along a straight path. Wheelchair direction commands were stored for observation and did not control the powered wheelchair. The AP walked beside the wheelchair,

along the AP path at approximately 1 m/s [64]. Ultrasonic tethering system performance was evaluated using tether distance performance parameters (mean absolute error, average, standard deviation) and wheelchair direction command errors (total number of left and right turn commands that should be forward commands since the wheelchair was controlled by the user).

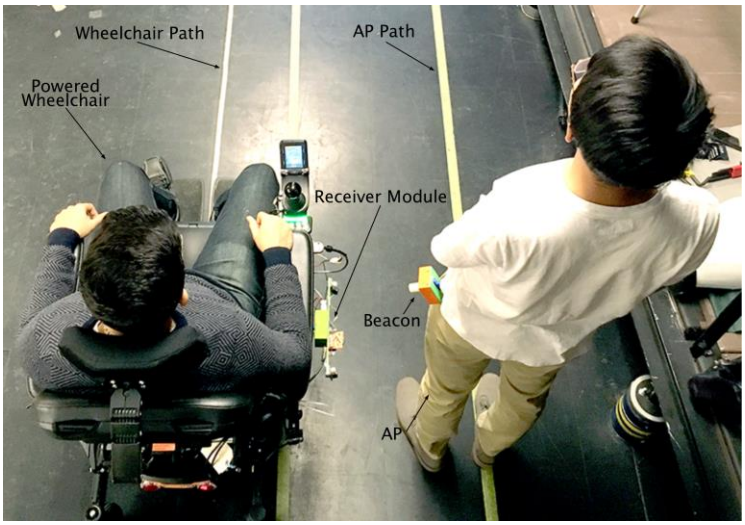
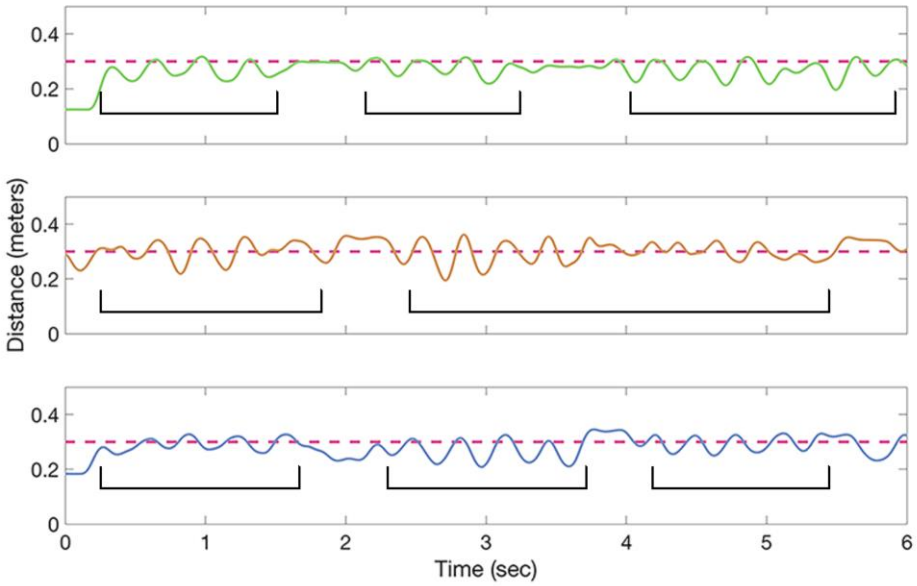


Figure 4.11 Ultrasonic tethering system performance test setup.

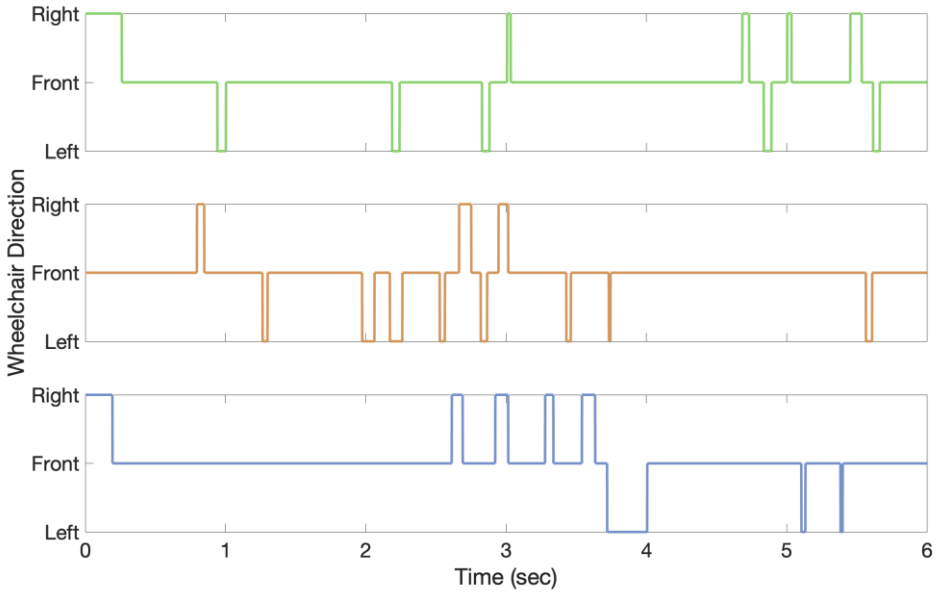
4.7 Results

The tethering system prototype was constructed using circuits and hardware designed during the simulation. The mounting and beacon frames were 3D printed using PLA. The piezoelectric transducers produced a noisy signal that had an amplitude change of approximately 0.07 V for every decimeter; therefore, a circuit was developed with a non-inverting amplifier (op-amp with gain 10) an active low pass filter (fcut = 48.2 Hz with 3dB attenuation), and demodulator using a diode based envelop detector. The signal required amplification with a gain of 10 to reduce clipping by the ATmega 328 based microcontroller's 10-bit analog to digital converter.

Figure 4.12 (a) shows tether distances for three trials acquired while a person walked beside the wheelchair, while the user controlled the wheelchair. The tether distance had a mean absolute error of 6 cm with a standard deviation of 7.6 cm. Figure 4.12 (b) shows the powered wheelchair direction commands for each trial acquired while the AP walked beside the wheelchair. Wheelchair direction commands were generated by a rule-based algorithm (Appendix C) that compared tether distance and tether angles to thresholds (Section 4.4.4). The thresholds and direction errors for the three trials are given in Table 4-2.



(a)



(b)

Figure 4.12 (a) Measured tether distance for the three walk trials. The dashed line represents the experimental tether distance (30 cm). The brackets indicate the effect of AP walking on the calculated tether distance, observed as sinusoidal waves, and (b) Wheelchair direction commands for the three walk trials.

Table 4-2 The low and high thresholds and the wheelchair direction command errors produced by the drive control for each trial.

Trial	Low threshold (m)	High threshold (m)	% Error
1	0.22	0.31	12.0
2	0.23	0.35	9.8
3	0.23	0.33	14.1

4.8 Discussion

Social following requires a powered wheelchair to follow beside a person at an appropriate conversational distance and location. This research successfully demonstrated that a prototype ultrasonic-tethering approach is viable for generating control signals to initiate and maintaining casual proxemic communication distance between the AP and powered wheelchair user. A real time plug-and-play ultrasonic tethering system was designed with a modified Eightfold Technologies SmartChair Remote as the interface between the "ultrasonic tethering components and custom navigation software" and the wheelchair control system.

The ultrasonic tethering system along with a SmartChair Remote can be connected to most commercially available powered wheelchairs, making the proposed solution broadly accessible to wheelchair users. The ultrasonic tethering system was developed with inexpensive components that were easily available in the market and all programming was implemented on an open source platform. The overall cost of developing the prototype was approximately $200 (CAD), which included the cost for manufacturing the 3-D printed parts (i.e., cases, clamps, holders), excluding the SmartChair Remote. Other contactless tethering techniques, such as using cameras and lidar sensors, require a computer with additional modules to convert digital signals to the required analog signals (and vice-versa) that are bulky and increase the overall system cost.

The ultrasonic system requires sensor calibration that stores and calculates the thresholds. These thresholds are based on a comfortable distance selected by the AP and the distance from the ground to the beacon. Sensor signals are affected by AP height, particularly the beacon height from ground level. To accommodate a range of AP heights, the prototype had hinges on the beacon and receiver modules to allow these components to rotate vertically and thereby provide a suitable sensor field-of-view. While this process would only take several seconds, the user may find the calibration phase and adjusting the beacon and receiver module angles as a hindrance.

Walking trial results showed that tether distance varied due to the AP non-parallel motion with respect to the wheelchair. AP not walking in a perfect straight line, or the wheelchair moving off center to the pre-determined path, caused the measured average tether distance to be lower than the actual tether distance. Tether distance error was also caused by left to right sway due to person's gait [65], where the range sensor detected the sway and output an oscillating signal that affected the calculated tether distance. This sinusoidal motion could cause an ultrasonic tethering system to change operating modes from available (tethered) to loss of tether or produce medial-lateral wheelchair undulations. In a social following control system, these deviations must be filtered, or removed by curve fitting and implementing a wheelchair trajectory planning algorithm [41] [66], to avoid wheelchair lateral undulated motion when matching AP movement. Safe and smooth wheelchair navigation is required for successful social following. Wheelchair kinematics may differ from the accompanying person's motion since the person can abruptly start, stop, and turn while walking. However, powered wheelchairs require smooth starts and stops, and soft turns for the user's safety and comfort. To achieve this, wheelchair trajectory planning and smoothing algorithms are needed for appropriate drive control [41], [66].

4.9 Conclusion

Determining the accompanying person's trajectory for social following poses a challenge to tethering-based wheelchair navigation. This research combined ping sensors and active/passive components with a shared wheelchair-control algorithm to make a plug-and-play contactless tethering device. This was achieved using commercially available ultrasonic sensors, a microcontroller, 3-D printed components, and an Eightfold Technologies SmartChair Remote.

Principles of ultrasonic ranging were used to determine an accompanying person's position and heading. A beacon was worn on the person's lateral waist and ultrasonic sensors on the wheelchair determined the accompanying person's pose by triangulation. Experimental results showed that the pose detection algorithm had a 6 cm tether distance error and the wheelchair direction error was 12%. The errors were due to pelvic sway during walking, and could be accommodated by modifying the thresholds and implementing wheelchair trajectory planning and smoothing algorithms. With these considerations, ultrasonic tethering can be a viable social following technology.

5 Smart Wheelchair Navigation using Ultrasonic Tethering for Social Following

5.1 Overview

This chapter discusses the ultrasonic tethering system software design and development and aids in addressing objective 1 and 2. This chapter discusses an improved wheelchair navigation algorithm implemented in the ultrasonic tethering software that achieves a constant and stable social following technique. The algorithms discussed in this chapter are shown in Appendix C.

5.2 Abstract

Social following for powered wheelchairs is described as the process of following beside an accompanying person (AP) during social situations. Conversations between the wheelchair user and the AP could lead to increased mental stress for the user and distractions from controlling or navigating the wheelchair, that could lead to accidents. Chapter 4 implemented social following on a powered wheelchair using ultrasonic tethering as a plug-and-play smart wheelchair system. Ultrasonic tethering determined the AP pose and thus determined wheelchair direction using a preliminary drive control algorithm. However, the algorithm was inaccurate because of factors ranging from different AP walking speeds and AP left to right sway while walking, which resulted in varying tether distance. This paper implemented a new drive control algorithm that aims to reduce tether distance variability by using a windowing technique to determine an appropriate wheelchair trajectory based on the AP trajectory, and maintained a constant conversation distance. The prototype was mounted on a Permobil F3 Corpus using a SmartChair Remote as an interface between the ultrasonic tethering system and the power wheelchair. The system was tested to determine the accuracy of identification and wheelchair direction control. The system successfully navigated the wheelchair based on the AP trajectory, maintaining an average user-selected conversation distance of 0.36 ± 0.1 m and controlled wheelchair direction with an error of 13%.

5.3 Introduction

Smart wheelchairs were developed to benefit people with mobility disorders by improving productivity and quality of life by increasing leisure and social participation [1]–[4]. Several human-following or tethering smart wheelchairs have been developed to follow behind a person, using technologies from mobile robots [43]–[45]. Smart wheelchair tethering provides hands-free wheelchair control while moving with a caregiver/guide/companion, enabling users to participate in social interactions or, for people with some locomotor capacity, providing access to the wheelchair while walking or exercising.

Most human-following smart wheelchair research implemented a follow-behind technique [43]–[45]. In social situations where the wheelchair user is accompanying another person, follow-behind would harm social interactions by making conversation more difficult and removing the ability to interpret facial cues, etc. Side-by-side following would be preferable to enable better social interactions.

Current human-following algorithms use the accompanying person's (AP) identification, pose, and velocity to determine the wheelchair trajectory [43]–[45]. Pose and velocity were determined from ping sensors, photoelectric sensors, cameras, or variable tension strings. Wheelchair trajectory was defined using mapping, localization, and path planning algorithms to determine the current state using information from exteroceptive and proprioceptive sensors. Only one system was reported for side-by-side following [47]. This system used an omni-directional camera and laser range sensors to detect and determine the person's pose. The system was capable of continuous person tracking while following but was unreliable due to environmental factors that affected the sensors.

To achieve side-by-side following, a novel powered wheelchair drive control concept was developed using an ultrasonic tethering (UT) smart add-on system. The UT system controlled wheelchair direction using a commercially available Eightfold Technologies SmartChair Remote [23]. Figure 5.1 shows the UT prototype mounted to the right armrest of a Permobil F3 Corpus powered wheelchair.

Figure 5.1 The ultrasonic tethering receiver mounted on a Permobil F3 Corpus wheelchair.

As reported in our previous research [53], a real time plug-and-play UT prototype was capable of identifying the AP and determining AP pose. The prototype used commercially available ping sensors, piezoelectric ultrasonic transducers, Atmel-based microcontroller, active/passive components (resistors, capacitors, diodes, operational amplifiers), and 3D printed models. The prototype was designed to:

- Identify the AP using a beacon on the AP and three piezoelectric ultrasonic receivers on the wheelchair's side (in the tether field-of-view)
- Determine AP pose by calculating the tether distance using two ping sensors placed on the wheelchair's side,
- Control powered wheelchair navigation by interfacing with a commercially available SmartChair Remote [23].

The prototype was evaluated by three walk trials, where the AP and wheelchair followed a pre-determined fixed straight path for six seconds while sensor data were acquired to calculate the tether distance. The results concluded that ultrasonic tethering is a viable approach for social

following since the UT prototype was capable of contactless tethering to an AP using triangulation, and generated wheelchair direction commands to initiate and maintain a casual communication distance between the AP and powered wheelchair user [52] [53]. However, high tether distance variability occurred, due to AP non-parallel walking motion with respect to the wheelchair, wheelchair moving off center to the pre-determined path, and AP left and right sway due to natural human gait [65]. The high tether distance variability caused a 12% error in wheelchair direction commands produced by the UT system [53].

To improve side-by-side following performance, an AP velocity algorithm and new wheelchair drive control concept are proposed and implemented to the UT prototype. This paper discusses the AP velocity algorithm and the wheelchair drive control concept that reduced the tether distance variability and provided smooth powered wheelchair navigation.

5.4 Identifying and determining the accompanying person's pose

The first process in powered wheelchair social navigation using ultrasonic tethering is identifying the AP, to avoid powered wheelchair navigational error from tethering to unwanted objects beside the wheelchair [53]. A beacon was attached to the AP waist (belt, pocket, etc.) and a receiver module (three piezoelectric receivers) was located on the wheelchair's side facing the AP. The beacon produced ultrasonic pulses at known intervals towards the wheelchair receiver module. The analog signals produced by the piezoelectric receivers were compared to thresholds to identify the AP in the receiver field of view. Determining AP pose, thereby determining tether distance and tether angle, was achieved by triangulation using two ping sensors [53] [60]–[62]. The AP velocity was determined by calculating the AP orientation with respect to the wheelchair. AP orientation was calculated using equation 4.5. During calibration, the AP orientation thresholds were calculated using equations 5.1 and 5.2.

$$\theta_{\text{low}} = \theta - \text{m} \tag{5.1}$$

$$\theta_{\text{high}} = \theta + \text{n} \tag{5.2}$$

where, θ_{low} and θ_{high} are low and high thresholds calculated during the sensor calibration phase, respectively (Section 4.5.5), m and n are threshold factors calculated during calibration, and θ is the AP orientation in degrees.

5.5 Sensor crosstalk and its effects on the identification algorithm

Crosstalk was observed when the beacon and the ping sensors were active at the same time. The beacon and the range sensors produce similar ultrasonic pulses (between 40 KHz to 42 KHZ) which caused interference and produced noisy and attenuated distance measurements from the range sensors. Figure 5.2 shows the effect of beacon and ultrasonic sensor crosstalk, by varying the distance from the sensor to the beacon. Sensor output (distance) was attenuated when the beacon was on between 0.00 and 0.65 meters.

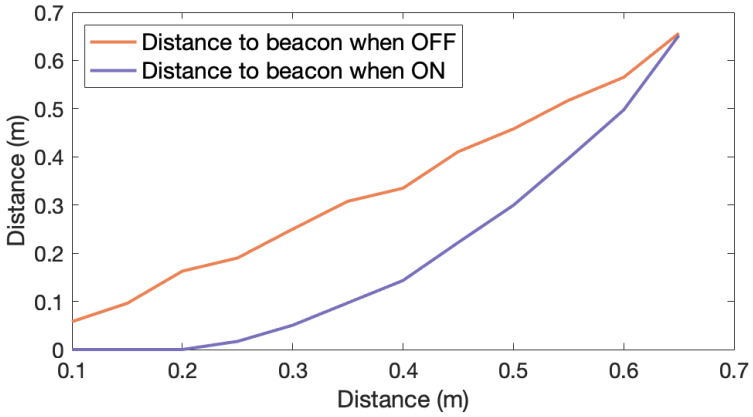


Figure 5.2 Comparison between distances from the receiver with the beacon on and off.

Eliminating crosstalk was necessary to accurately determine AP pose and velocity. Identification only occurs when the beacon is switched on, in the tether field-of-view. Two algorithms can help eliminate crosstalk with identification.

5.5.1 Identification algorithm I

A simultaneous identification function was implemented to time-delay synchronous on and off functions to the receiver and the beacon. The beacon switched on and off at fixed intervals, with these intervals calibrated based on the overall time required for the receiver's software to run multiple iterations. The beacon's on state was 100 ms, to reduce the time when the receivers were switched off. During this time, no distance information was collected to calculate AP pose and velocity. The beacon's off state was 1020 ms. The receiver measured AP distance from the two ping sensors (L, T) during the off state and identified the AP during the on state. Figure 5.3 shows the on and off times intervals for the receiver and the beacon.

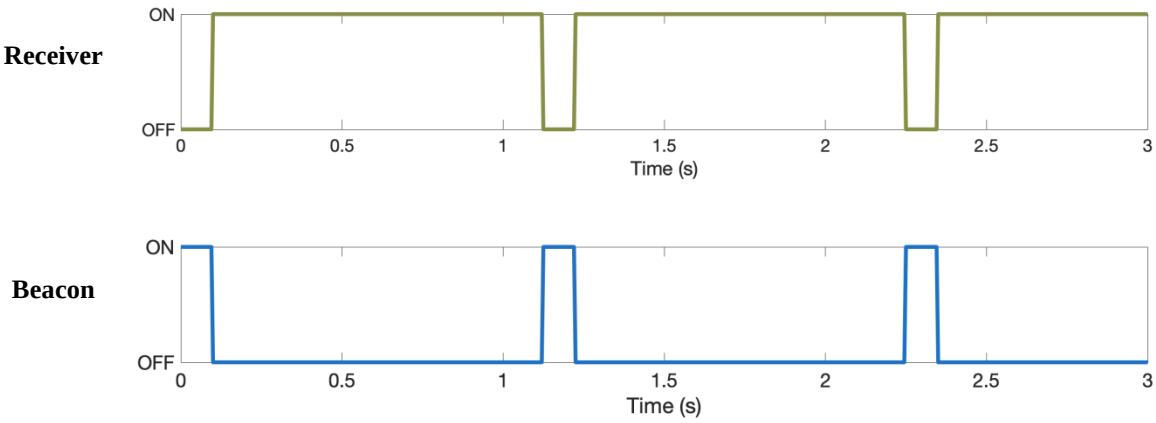


Figure 5.3 Receiver and beacon on, and off time intervals (states).

5.5.2 Identification algorithm II

The second identification algorithm included non-continuous and manual AP identification function, where the AP was identified once (after sensor calibration) until the AP was outside the tether field-of-view. The UT system was designed to 'search' for the AP until the AP manually switched on the beacon. The UT system shifted into tethering mode after the AP was identified and produced a tone on the buzzer to inform the AP to turn off the beacon. AP un-identification occurred when the AP moved outside the tether field-of-view for 1 second. The AP switched on the beacon for re-identification.

5.6 Wheelchair drive control concept

The wheelchair drive control concept was developed to reduce the tether distance variability that caused wheelchair oscillations that would cause uncomfortable motion for the wheelchair user. To reduce these oscillations, the concept included tether distance filtering and wheelchair direction smoothing. The wheelchair drive control design criteria are:

- Determine wheelchair navigation based on the AP pose and velocity
- Control wheelchair direction through an Eightfold Technologies SmartChair Remote [23]
- Provide smooth powered wheelchair navigation
- Must have minimal effect on the UT system onboard microcontroller memory.

Based on the design criteria, the drive control included a wheelchair trajectory and speed algorithm and a navigation control algorithm. The trajectory algorithm determined the appropriate wheelchair trajectory based on the AP trajectory. The navigation algorithm determined the wheelchair direction control commands, to maintain a comfortable conversation distance during social following. Wheelchair kinematics differed from the AP motion since the AP can start, stop, and turn abruptly while walking. The navigation control algorithm included a smoothing function for soft starts, stops, and turns, for user safety and comfort.

5.6.1 Wheelchair trajectory and speed algorithm

Navigating the wheelchair beside the AP is achieved by calculating the AP's trajectory and determining the appropriate wheelchair trajectory [41], [67]. The overall AP path consisted of smaller multiple paths (windows). The window size was 340 ms, based on the maximum time required for the software to run one iteration and minimum time required for the AP to change direction. Each 340 ms smaller path had fixed directions, forward, right, and left (Figure 5.4) [41].

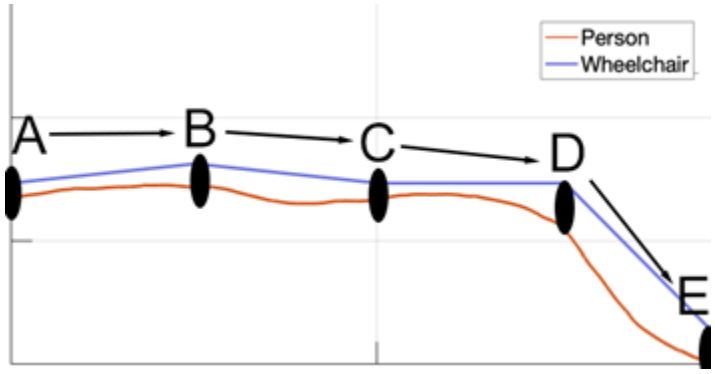


Figure 5.4 Representation of an overall path between the starting position (A) and the destination (E), with multiple smaller paths of fixed directions (A to B; B to C; etc.).

The ultrasonic tethering system determined the AP trajectory based on the AP pose and heading. The algorithm to determine wheelchair direction based on AP trajectory is shown below:

1. At the start of tethering, the wheelchair was programmed to move forward in a straight line for a window of 340 ms. During this time all ping sensor data were recorded and stored in an array for that window. Sensor data were filtered using

equation 5.3, where L, T are the distances from the front and back sensor to the AP (Figure 4.7), w is the window number, n is the number of iterations. The first-order low pass filter had a 1.7 Hz cut-off frequency, based on AP walking speed.

$$\begin{bmatrix} T_i \\ L_i \end{bmatrix}_w = \begin{bmatrix} \frac{1}{n} * \sum_{i=0}^{n} 0.8987 * T_i + 0.1013 * T_{i-1} \\ \frac{1}{n} * \sum_{i=0}^{n} 0.8987 * L_i + 0.1013 * L_{i-1} \end{bmatrix} \quad \textbf{(5.3)}$$

2. The tether distance (D) for each iteration (i) during the first wheelchair motion $(w = 1)$ was calculated using equation 5.4 [53], where D is the tether distance, L and T are the filtered distances and M is the distance between sensors (24 cm). AP heading was calculated using equation 4.5, where θ is the heading and α and β are the angles formed between L and T and the AP.

$$D_i = \sqrt{\frac{L_i^{\,2} + T_i^{\,2}}{2M}} \quad \textbf{(5.4)}$$

3. The tether distance and heading were filtered and stored in an array for smoothing, and were calculated using equation 5.5, where, w is the window number, n is the number of iterations and, K_{d1} and K_{d2} are the filter gains (0.8987, 0.1013), calculated based on a low-pass filter with cut-off frequency at 1.7 Hz.

$$\begin{bmatrix} D_i \\ \theta_i \end{bmatrix}_w = \frac{1}{n} * \sum_{i=0}^{n} \begin{bmatrix} K_{d1} * D_i + K_{d2} * D_{i-1} \\ K_{d1} * \theta_i + K_{d2} * \theta_{i-1} \end{bmatrix} \quad \textbf{(5.5)}$$

4. AP orientation was used to determine the required wheelchair speed. The wheelchair speed based on the AP relative velocity was divided into three speeds (30%, 50%, 70% of user selected maximum speed from the joystick settings, Section 5.6.5). The wheelchair speed was determined using equation 5.6, where θ_{low} and θ_{high} are low and high thresholds calculated using equations 5.1 and 5.2.

$$[\text{speed}]_w = \begin{cases} 1, & \theta_i < \theta_{low} \\ 2, & \theta_{low} < \theta_i < \theta_{high} \\ 3, & \theta_{high} < \theta_i \end{cases} \quad \textbf{(5.6)}$$

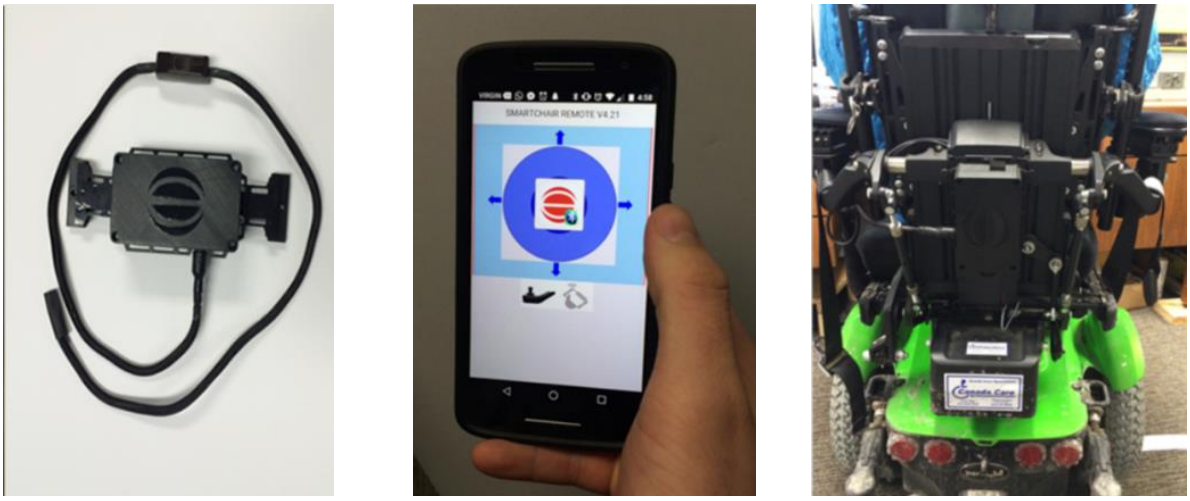
A new wheelchair trajectory was determined using the smoothed tether distance in the previous window ($w = w - 1$). The algorithm compared the previous tether distance curve to precomputed curve thresholds to determine if the AP was moving in a straight line, a right curve or a left curve, or stopped walking. Steps 2, 3, and 4 were repeated for each window until the system was turned off or the AP became untethered.

5.6.2 Wheelchair navigation control algorithm and the SmartChair Remote

The navigation control algorithm was used to control the wheelchair direction based on the wheelchair commands produced by the wheelchair trajectory and AP velocity algorithms. The navigation control algorithm produced smooth changes in direction commands that provide soft starts and stops, and smoothed curved turns. Soft starts and stops are required to maintain comfortable navigation for the user. Sometimes, the wheelchair direction commands produced by the trajectory algorithm changed abruptly at the end of one window, causing sporadic wheelchair motion. The sporadic motion was reduced by iteratively reducing or increasing the wheelchair speed to match the new direction.

5.6.3 SmartChair Remote

Powered wheelchair navigation was achieved by using an Eightfold Technologies SmartChair Remote [23]. The SmartChair Remote was developed as a Bluetooth interface between a Smartphone and any commercially available powered wheelchair. Powered wheelchair navigation can be achieved remotely using a Smartphone application, where the user remotely moves the powered wheelchair using a graphical joystick on the application (Figure 5.5) [23]. The SmartChair Remote is advantageous because of its capability of providing seamless wheelchair control changes between the onboard joystick, Bluetooth controller, and UT system.



(a) **(b)** **(c)**

Figure 5.5 (a) The SmartChair Remote, (b) the SmartChair Remote smartphone app that can be used to control wheelchair direction, and (c) SmartChair Remote mounted to a powered wheelchair [23].

The navigation control algorithm transmits wheelchair direction commands to the SmartChair Remote using serial communication. The UT system directly connects to the Remote's power and control connector that in turn connects to the SmartChair Remote's inbuilt controller's serial port. This required minimum modification to the SmartChair Remote. A new function was added to the SmartChair Remote code to serially read data from the UT system. To reduce major SmartChair Remote software modifications, the UT system drive control was designed to transmit data using the same format as the data transmitted by the Smartphone Bluetooth module to control wheelchair direction, this also reduced hardware modifications to the commercially available SmartChair Remote.

5.6.4 User selected powered wheelchair speed settings

The wheelchair user can select a preset maximum wheelchair speed using the speed selector on the joystick controller (Figure 5.6), that ranges from speed setting 1 to 5. The UT system's navigation control algorithm transmits a speed command (1, 2, or 3) to the SmartChair Remote based on the AP velocity. Therefore, wheelchair speed control is only possible from 0 to 100% of the user selected speed.

(a) **(b)**

Figure 5.6 (a) Speed 1 (low speed) setting and, (b) Speed 5 (high speed) setting and the wheelchair speed selector interface on the manual joystick controller.

5.6.5 Wheelchair navigation – direction and speed commands

The wheelchair navigation commands included concatenating a direction command with a speed command (Table 5-1). Commands were produced by the drive control algorithm's loop function. For example, when required to move the wheelchair forward for 1 window with a speed of 50% of the maximum user selected speed, the wheelchair command was "*f2*".

Table 5-1 (a) UT system direction commands with the wheelchair direction, and (b) Speed commands with the wheelchair speed (in % of the maximum user selected speed on the joystick).

(a)

Direction Command	Wheelchair Direction
f	Forward
s	Stop
l	Left
r	Right

(b)

Speed Command	Wheelchair Speed (%)
1	30
2	50
3	70

5.6.6 Hard and soft turns using the SmartChair Remote

Most powered wheelchairs use a differential drive system (i.e., two motorized wheels and two free moving castors to control move forward, backward, left and right) [68]. The differential drive system can be used to turn or make complete 360 degree in-place rotation (pivotal steering or zero-radius turning) [24]. Hard turns are zero-radius turning and soft turns are left and right turns along an arc. The difference between executing a hard and soft turn in powered wheelchairs using a joystick is shown in Figure 5.7. Hard turns can be executed by moving the joystick to the east or west positions, whereas, soft turns can be executed by moving the joystick to the north-east or north-west positions.

(a) **(b)** **(c)** **(d)**

Figure 5.7 Wheelchair directions by moving the joystick to the (a) hard right (east), (b) soft right (north-east), (c) hard left (west), and (d) soft left (north-west).

The SmartChair Remote was designed to emulate a physical joystick by producing signals equivalent to the joystick positions. Therefore, to reduce software and hardware modifications to the SmartChair Remote and to successfully execute soft turns using the SmartChair Remote, the UT system's navigation algorithm produces a forward command after every hard right or left command. The joystick emulator converts and modulates the rapid change in direction from the UT system to soft turns.

5.7 Ultrasonic tethering system testing

The ultrasonic tethering system was mounted under the armrest to the right of a Permobil F3 Corpus powered wheelchair and connected to an Eightfold Technology's SmartChair Remote (Figure 5.8). Ultrasonic tethering testing included comparing the wheelchair commands produced by the UT system to the AP direction. Two tests were designed to study and determine the powered wheelchair's direction accuracy based on the AP trajectory.

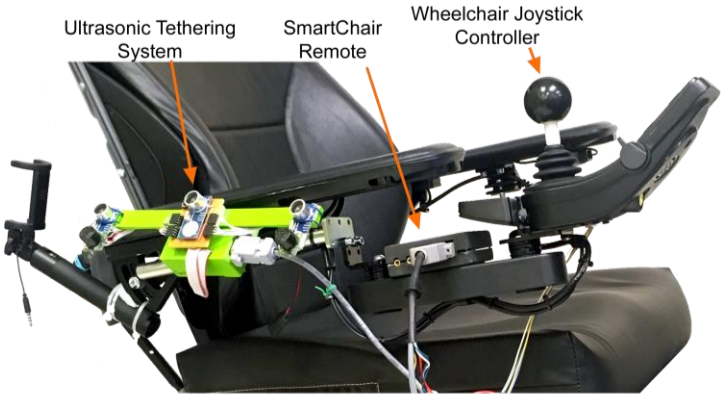


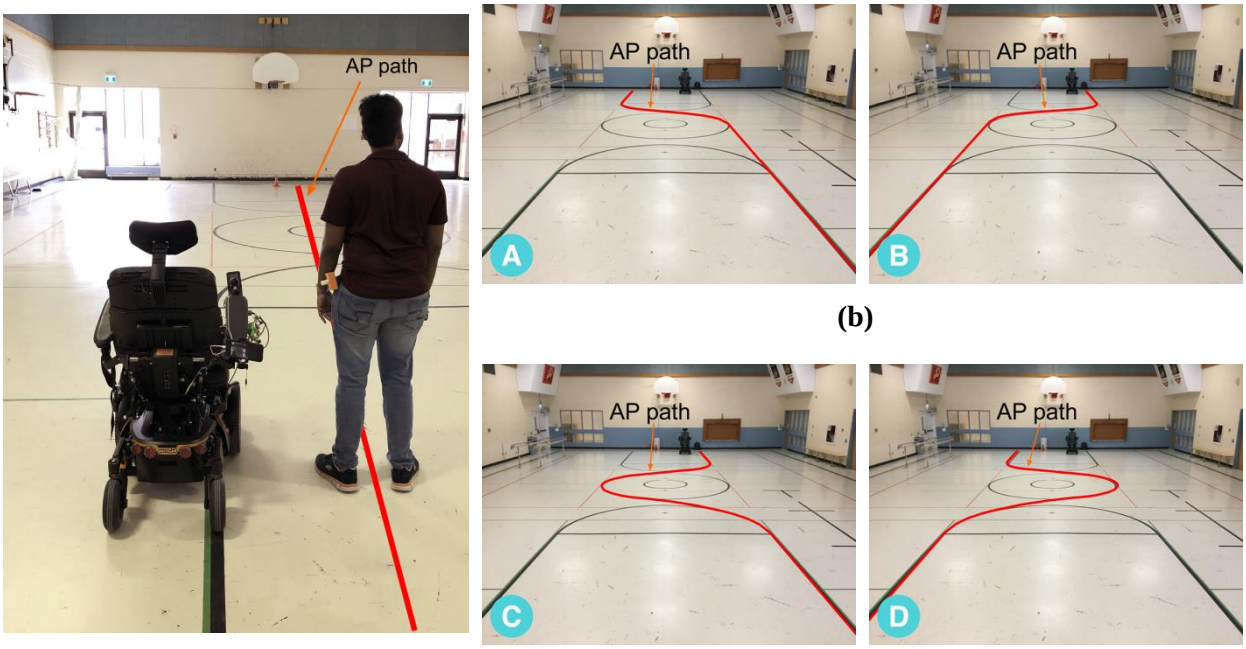
Figure 5.8 Ultrasonic tethering system, the SmartChair Remote and the joystick controller.

5.7.1 Straight path test

The straight path test involved the AP walking on a straight path for approximately 26 meters, for a total of 10 trials, with the wheelchair following beside the AP using the UT system. This test evaluated the wheelchair drive control concept's effectiveness in reducing non-parallel motion between the AP and the wheelchair. The straight path test was also designed to check the tethering accuracy and ability to maintain appropriate conversation distance between the wheelchair and the AP. The test setup is shown in Figure 5.9. The straight path test also evaluated the first identification algorithm (Section 5.4.2), where the UT system continuously identified the AP for every 1.02 seconds.

5.7.2 Mixed path test

The mixed path test included AP walking along straight and curved paths with the wheelchair following beside the AP using the UT system. Four types of paths were used, incorporating different configurations of curved and straight paths (Figure 5.9).



(a) **(b)** **(c)**

Figure 5.9 (a) The straight path test setup, (b) mixed path A and B, (c) mixed path C and D.

The mixed path test also evaluated the second identification algorithm. The directional error for each test was calculated using equation 5.7 where, CMD_{UT} is the number of direction commands (*f, l, r, s*) produced by the UT system to maintain the desired path and, Dir_{AP} is the actual AP direction.

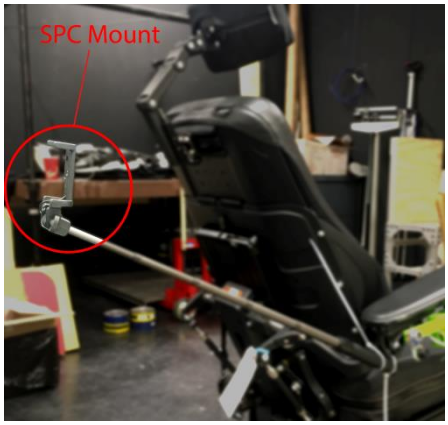
$$\text{direcional error (\%)} = \frac{\sum \text{CMD}_{\text{UT}} - \sum \text{Dir}_{\text{AP}}}{\sum \text{Dir}_{\text{AP}}} * 100 \qquad \textbf{(5.7)}$$

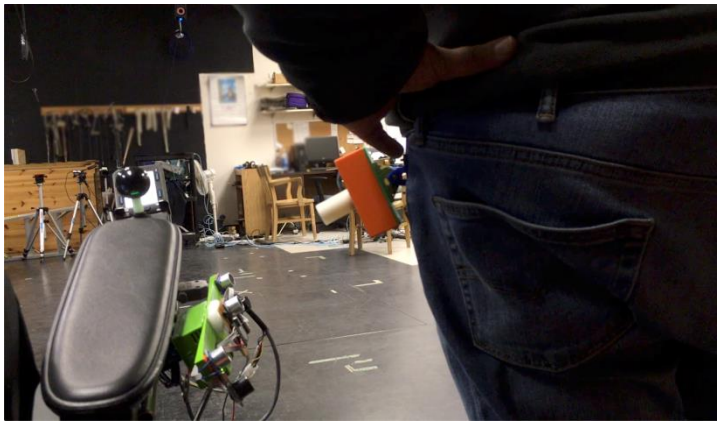
All trials included an initial calibration phase, where the AP stood next to the wheelchair for five seconds, in the tether field-of-view, at a comfortable distance for conversations between the AP and the wheelchair user. The UT system read all sensor data and calculated the required tether distance and tether angle for thresholds.

All trials were conducted in the gymnasium at the Ottawa Hospital Rehabilitation Center (TOHRC), the approximate dimensions for the gymnasium are 28 m long and 15 m wide (420 m^2). Sensor data, tether distance, tether angle, wheelchair direction commands, and calculated AP position were collected for each test (10 trials). Additionally, video was recorded using a

Smartphone camera (SPC) placed on a fully extended Smartphone mount (selfie-stick) behind the powered wheelchair's right-side armrest, facing towards the AP and wheelchair (Figure 5.8 and Figure 5.10). The video was used to check if the wheelchair responded accurately. Identification accuracy was calculated by counting the number of times the wheelchair moved in such a way that the AP moved outside the video frame. A sample image of the video is shown in Figure 5.10.

(a)





(b) **(c)**

Figure 5.10 (a) UT system testing area at TOHRC Gymnasium (panoramic view), (b) Smartphone camera (SPC) mount, and (c) SPC video screenshot.

5.8 Results

The UT system software was implemented on an ATmega328 microcontroller using an open-source platform and a compiler. The UT system software used 42% of the program storage space, and 74% of dynamic memory allocated for the global variables.

5.8.1 Calibration results

A total of 20 calibrations were performed (one before each trial). The AP stood next to the wheelchair at an average tether distance of 0.36 m and a tether angle of -4 degrees. Using equation 4.1, the mean conversation distance between the AP and the wheelchair user was 1 m.

5.8.2 Straight path test results

Figure 5.11 shows screenshots from two trials during the straight path test.

Figure 5.11 Processed video screenshots from the straight path test collected from a fixed camera (main video) and the smartphone camera mounted to the wheelchair (top right video overlay).

Figure 5.12 shows tether distance during walking (purple), tether distance during the initial calibration phase (red), low and high thresholds (red bounds) and the direction commands produced by the UT system to control the wheelchair direction for ten straight path trials.

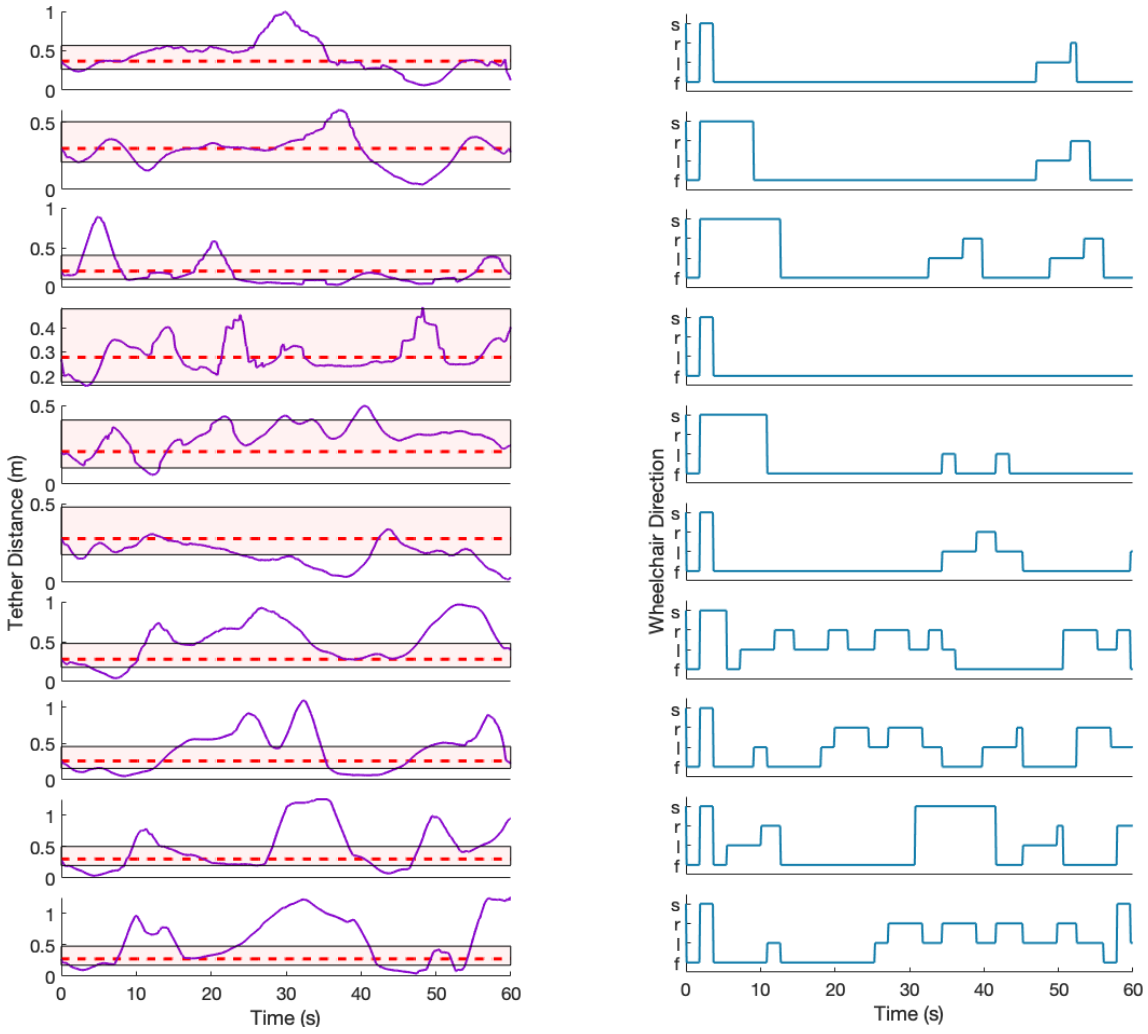
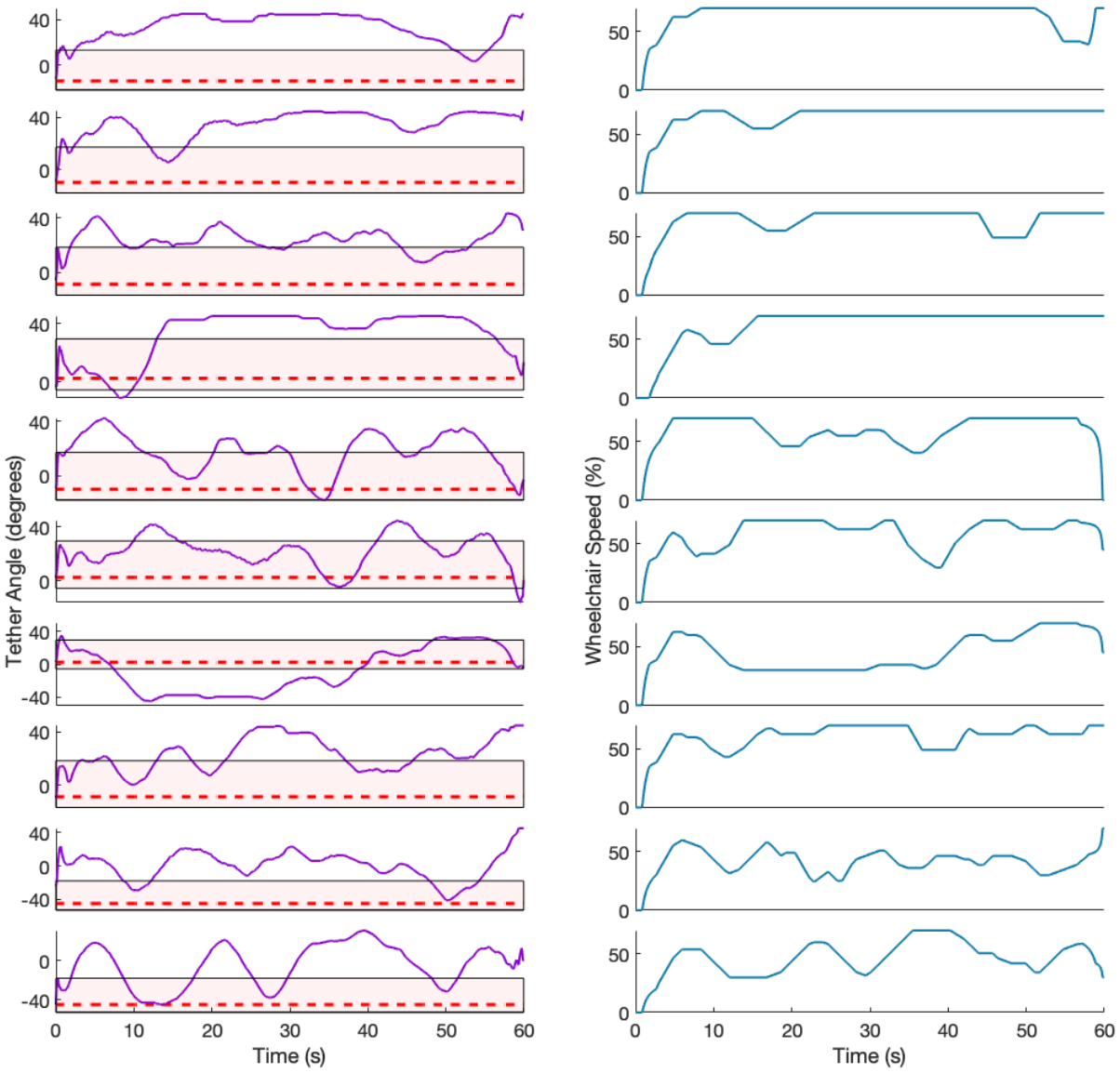


Figure 5.12 Straight path test results for 10 trials. (a) AP tether distance during walking (purple), Precomputed tether distance (dashed red), threshold band (low and high threshold calculated during calibration, red band) and, (b) tether distance corresponding wheelchair direction commands (f – Forward, l – Left, r – Right, s – Stop) produced by the UT system for each trial.

Figure 5.13 shows the tether angles, tether angle during the initial calibration phase (red), low and high thresholds (red bounds) and the calculated wheelchair speed (% of the maximum user selected speed). Speeds were output from the SmartChair Remote, and correlate to the speed commands generated by the UT system (i.e., (1=30%, 2=50%, 3=70%). The SmartChair Remote smoothed transitions between the discrete speed commands, creating curves in Figure 5.13 (b).



(a) **(b)**

Figure 5.13 Straight path test results for 10 trials. (a) AP orientation during walking (purple), Precomputed tether angle (dashed red), threshold band (low and high thresholds calculated during calibration, red band), and (b) wheelchair speed (0 – 70% of the user selected speed).

Table 5-2 shows the overall test outcomes; including, tether distance and angle during calibration, direction commands produced by the UT system (in % of the overall commands produced by the UT system for each trial), and directional error.

Table 5-2 Straight path test results: the tether distance threshold (T-D) and tether angle (T-A), wheelchair direction commands (F – forward, L – left, R – right, S – stop) in percent of all commands transmitted to the SmartChair Remote (using equation 5.7).

Trial	T-D (m)	T-A (deg)	F (%)	L (%)	R (%)	S (%)	Error (%)
1	0.36	-14	87.8	7.8	1.4	3.0	9.0
2	0.3	-10	75.0	7.9	4.6	12.5	14.7
3	0.2	-8	57.7	15.4	8.7	18.2	26.5
4	0.28	2	96.8	0.0	0.0	3.2	0.0
5	0.2	-10	79.0	6.0	0.0.	15.0	18.0
6	0.28	2	78.3	14.3	4.4	3.0	12.4
7	0.28	2	30.6	33.2	30.2	6.0	9.3
8	0.25	-8	45.2	27.4	24.4	3.0	7.5
9	0.3	-12	54.2	15.3	9.3	21.2	30.0
10	0.28	-16	39.2	29.0	25.6	6.2	7.2

5.8.3 Mixed path test results

Figure 5.14 shows screenshots from one trial during the mixed path test.

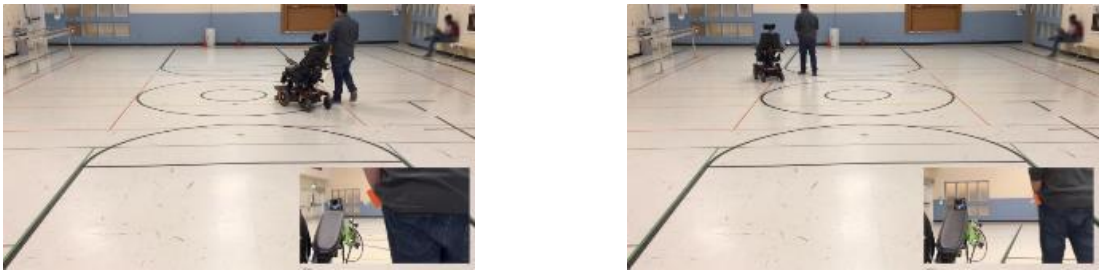
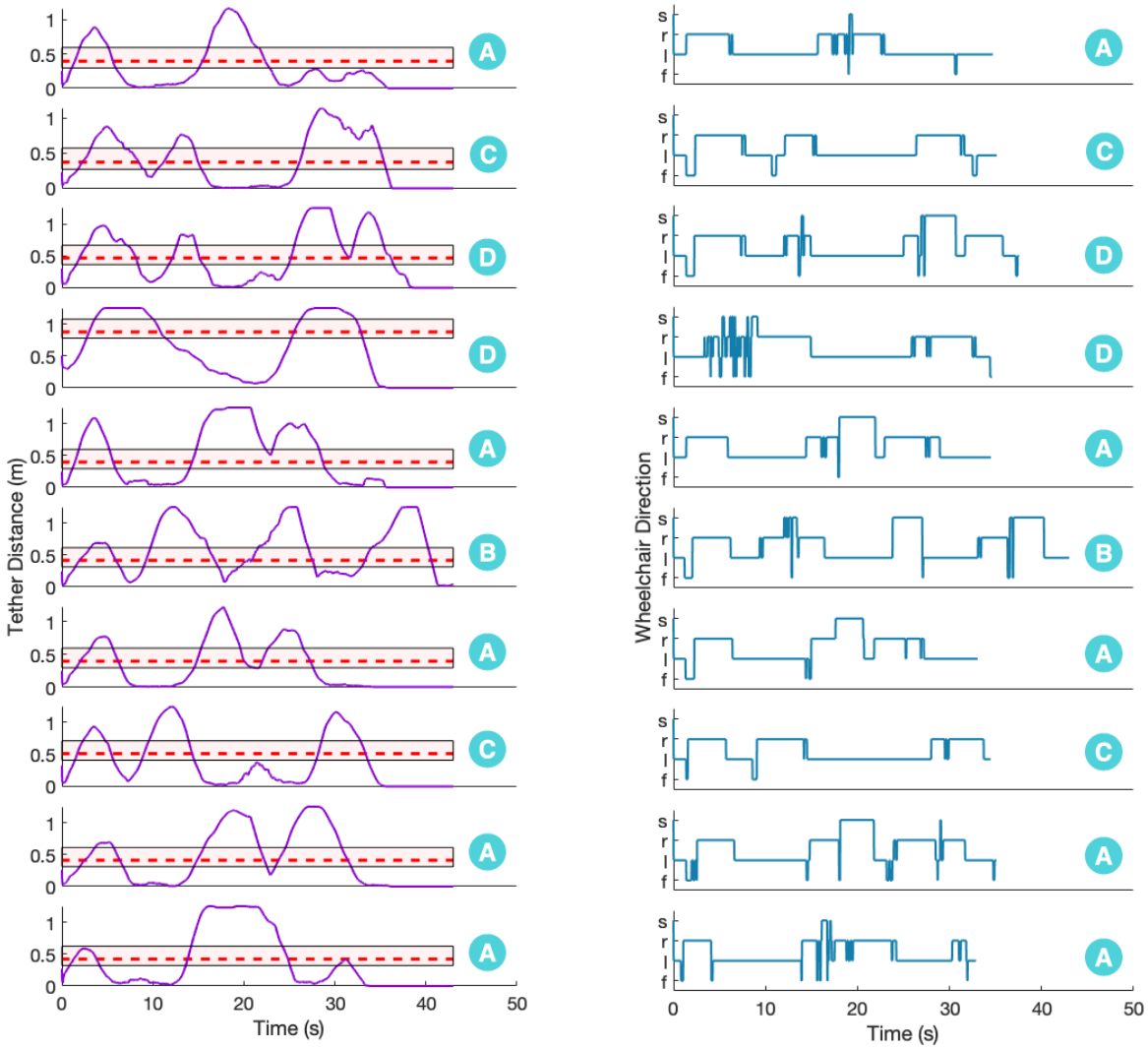


Figure 5.14 Processed video screenshots from the mixed path test collected from a fixed camera (main video) and the smartphone camera mounted to the wheelchair (bottom right video overlay).

For the mixed path test, Figure 5.15 shows tether distance during walking (purple), tether distance during the initial calibration phase (red), low and high thresholds (red bounds) and the direction commands produced by the UT system to control wheelchair direction.



(a) **(b)**

Figure 5.15 Mixed path test results. (a) AP tether distance during walking (purple), Precomputed tether distance (dashed red), threshold band (low and high threshold calculated during calibration, red band), (b) wheelchair direction commands (f – Forward, l – Left, r – Right, s – Stop) produced by the UT system. The four paths are labeled as A to D beside each graph.

Figure 5.16 shows the tether angles (in purple), tether angle during the initial calibration phase, low and high thresholds and the calculated wheelchair speed (in % of the user selected speed) (in blue) while walking, for ten trials acquired while the AP walked in a straight line. As with Figure 5.13, speeds were output from the SmartChair Remote and correlate to the speed commands generated by the UT system.

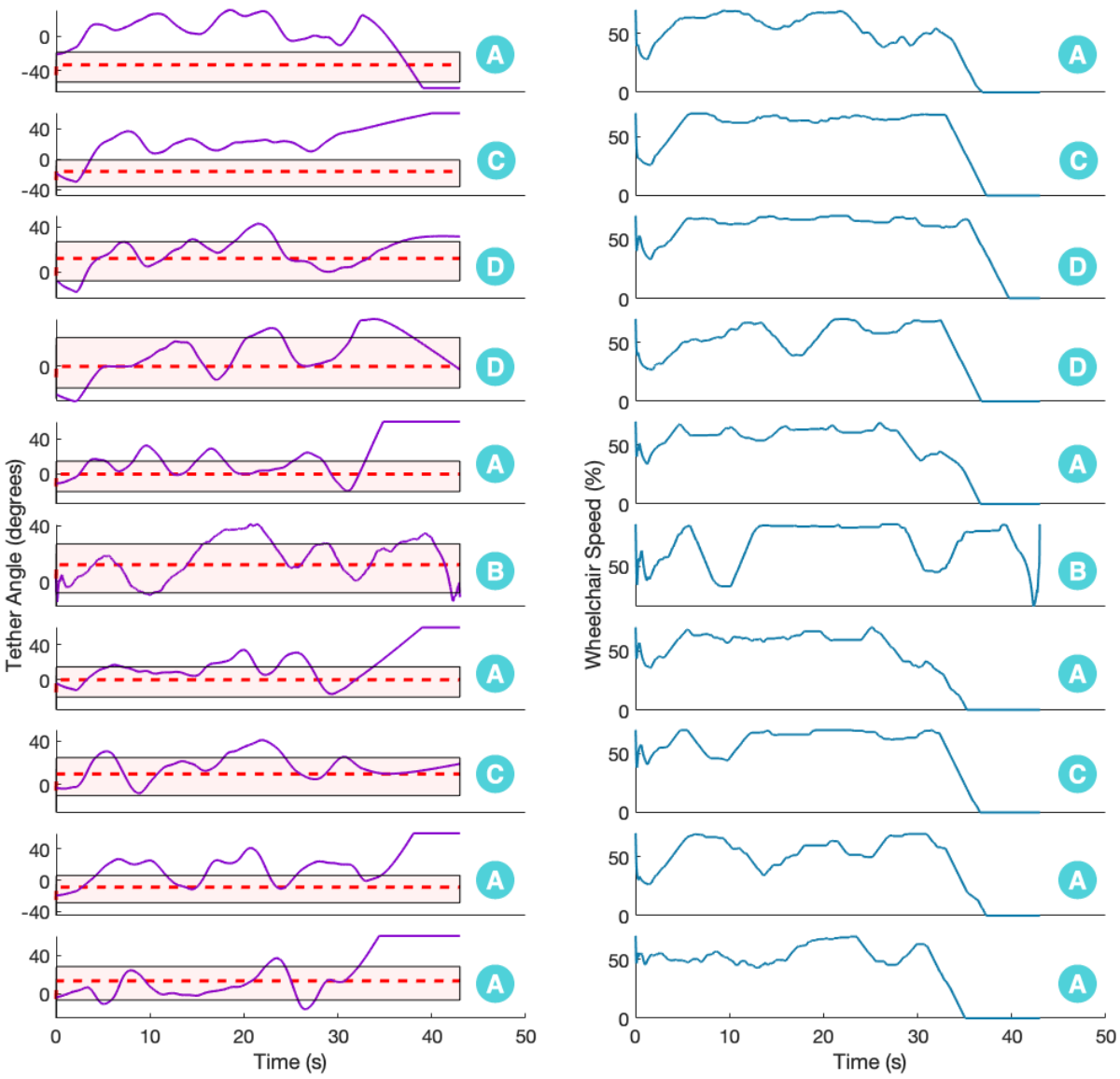


Figure 5.16 Mixed path test results. (a) AP orientation during walking (purple), Precomputed tether angle (dashed red), threshold band (low and high thresholds calculated during calibration, red band) and, (b) wheelchair speed (0 – 70% of the user selected speed). The four paths are labeled as A to D beside each graph.

Table 5-3 shows results for the tether distance and angle during calibration, identification error, and the AP paths used for each trial.

Table 5-3 Mixed path test results: The AP paths used for each trial, the precomputed tether distance (T-D) and tether angle (T-A) used for calculating the wheelchair's trajectory, and the number of times the system produced an identification error.

Trial	Path	T-D (m)	T-A (deg)	Identification Error
1	A	0.4	-32	0
2	C	0.38	-16	0
3	D	0.46	12	1
4	D	0.88	0	0
5	A	0.4	0	0
6	B	0.42	12	1
7	A	0.4	0	0
8	C	0.5	10	0
9	A	0.4	-8	1
10	A	0.42	12	0

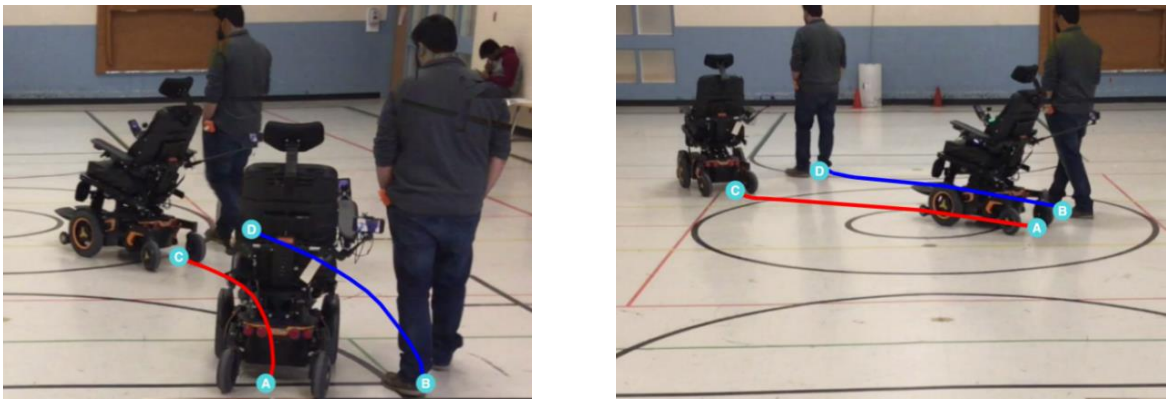
5.9 Discussion

The UT system software was implemented on an Atmel ATmega328 based microcontroller. The microcontroller has a maximum of 30720 bytes for storing the program and a maximum of 2048 bytes for dynamic memory (local variables). The overall UT system software used 42% of the program storage memory and 74% of the dynamic memory. Although sufficient memory is available for the program, dynamic memory may be insufficient for long UT system use. Dynamic memory stores local variable data, and, if insufficient, the system produces calculation errors. Low memory availability limited the overall UT system capability. Although, microcontrollers with higher memory are commercially available, the ATmega328 controller was at the brink of being sufficient for the prototype developed in this thesis.

The straight path test evaluated UT system accuracy for determining AP pose and velocity, thereby producing wheelchair direction commands to control the powered wheelchair's direction. The UT system produced direction commands (f-forward, l-left, r-right and s-stop) based on the

AP direction for each window. For straight walking, tether distance variability was reduced by implementing a window filter and curve fitting technique using a wheelchair trajectory algorithm; however, the UT system produced successive left and right commands for a few trials. The wheelchair navigation control algorithm produced a left command after producing a right command (or vice versa), to maintain constant conversation distance between the wheelchair and the AP. The process of producing a left command after producing a right command, or vice versa, maintained a conversation distance of 1 ± 0.35 m but produced smooth wheelchair undulations that could sometimes be uncomfortable for the wheelchair user.

The mixed path test evaluated UT system accuracy in following the AP along turns. The results showed that the UT system responded to AP left and right curved turns; however, the UT system was designed to turn the wheelchair along a fixed radius curve. Fixed radius turning occurred because the UT system's drive control algorithm could produce only four commands to control the wheelchair direction, with each direction command moving the wheelchair in a fixed trajectory. Fixed radius turning for the wheelchair affected AP gait while turning. To maintain tether, the AP had to walk a greater distance while turning left and less distance while turning right. In addition, the AP had to walk faster (left) or slower (right). As shown in Figure 5.17, the AP had to walk a distance BD to make sure that the wheelchair moved distance AC. Fixed radius turning limited AP motion beside the wheelchair. This limitation could be inconvenient for the AP and require the AP to be mindful of the wheelchair's motion.



(a) **(b)**

Figure 5.17 (a) AP turning left along BD and the wheelchair following the AP along AC, (b) AP turning right along BD and the wheelchair following the AP along AP.

The UT system controlled the powered wheelchair's speed based on the AP angle to the UT system (tether angle). The speed control algorithm produced wheelchair speed commands between 0 to 100% of the user selected speed on the joystick controller. The UT system controlled wheelchair speed to match the AP's speed.

The straight path test included testing the first identification algorithm, where the UT system continuously and cyclically identified the AP every 1.02 seconds. The straight walk test results showed identification errors from beacon switching on and off that did not correspond to the receiver's synchronized off and on. This could occur because the clocks used by the receiver's and beacon's microcontrollers could drift, thus causing asynchronous switching. Furthermore, the unsynchronized beacon and receiver on and off sequences resulted in short overlaps of the beacon's and receiver's on state. These overlaps caused sensor crosstalk that produced tether distance errors and therefore wrong wheelchair direction commands.

The straight path test results showed that the AP was untethered (unidentified) during trials 2, 3, 5, 7 and 9. During these trials, the AP walked forward faster than the UT system's window, thereby walking away from the tether field-of-view, as shown in Figure 5.18. The UT system produced the stop command because the AP was not tethered.

(a) **(b)**

Figure 5.18 Screenshots of the straight walk test during unidentification, where (a) the AP walked faster than the UT system's window thus causing an identification error, (b) the AP had to take a step back to be re-identified.

The mixed path test included testing the second identification algorithm, where the AP manually controlled switching the beacon on and off to avoid sensor crosstalk. Although, identification algorithm II was advantageous over algorithm I in avoiding sensor crosstalk, the AP had to manually control the beacon, which could be a hindrance.

5.10 Conclusions

A smart wheelchair add-on prototype was developed using ultrasonic tethering techniques that could be used for social following. This research presented a wheelchair trajectory control algorithm based on AP direction and speed of motion. Controlling the wheelchair motion was achieved by using a SmartChair Remote. Trial results showed that the UT system maintained a mean tether distance of 0.36 meters and produced a directional error of 13%. Trial results also showed that identification algorithm II performed better than identification algorithm I.

6 Thesis Conclusions and Future Work

This thesis designed and developed a human-following smart wheelchair system that used ultrasonic tethering to maintain a comfortable conversation distance between the wheelchair user and the accompanying person. The overall ultrasonic tethering system consisted of an AP identification algorithm, pose detection algorithm, AP velocity algorithm that accurately determined the accompanying person's pose and velocity, wheelchair trajectory algorithm, and a wheelchair drive control system that determined wheelchair direction commands to navigate the wheelchair at a distance beside the accompanying person. The AP can be identified beside the wheelchair from 0.2 m to 2 m, with a resolution of 0.02 m. The wheelchair trajectory algorithm accurately produced wheelchair direction commands ('f'-forward, 'l'-left, 'r'-right, 's'-stop) based on the AP trajectory. The drive control system controlled the wheelchair direction by transmitting direction commands to the Eightfold Technologies' SmartChair Remote [23]. Conclusions for each thesis objective are presented below.

Objective 1: Develop an ultrasonic tethering prototype to enable side-by-side following in social situations

An ultrasonic tethering prototype was developed using ultrasonic sensors, active and passive electrical components, and 3-D printed parts.

Objective 2: Test and validate the ultrasonic tethering prototype for smooth hands-free powered wheelchair navigation

The ultrasonic tethering prototype was tested for smooth hands-free powered wheelchair navigation. The system responded sufficiently accurately, but with an average error of 13 %. The error corresponded to wrong wheelchair direction based on actual AP direction. The results demonstrated that the AP must be consciously aware of the wheelchair, which helps to smooth wheelchair navigation for the user.

6.1 Future work

The research presented in this thesis is a novel technique for human-following using a contactless ultrasonic tethering system. Contributions of this thesis include a viable, cost-effective solution for side-by-side human-following using a sophisticated pose detection and navigation

algorithm with minimum computational power. The successful ultrasonic tethering system with the improved pose detection and navigation algorithm opens the door for improvements and many new research opportunities for assistive human-following smart wheelchairs.

6.1.1 Improved trajectory and speed algorithm

The current control algorithm determined AP pose for a fixed window (w = 340 ms). A 340 ms window size limited the ability to detect AP left and right turns and created a delay in producing wheelchair direction commands that responded to turns. An improved trajectory and speed algorithm could be studied that implements a shorter window size to reduce wheelchair directional errors. Improvements include using AP real-time speed to determine the optimal window size, which could be beneficial in detecting AP left and right turns within the calculated window and reduce the delay between the AP motion and wheelchair's motion.

6.1.2 Drive control for backward navigation and sharp left and right turns

The current drive control algorithm produces forward, left, right and stop direction commands. Implementing backward direction would be beneficial for a robust human-following system, and requires an additional computation step in the AP pose detection algorithm. In social situations, it may be necessary to navigate backwards to avoid obstacles directly in front of the wheelchair. The current UT system is designed for soft left and right turns. Powered wheelchairs have a differential drive system that allows hard turns (pivotal or zero-radius turns). Zero-radius turning would be beneficial for turns along a sharp path. However, for continuous contactless tethering, hard turns may cause tethering issues because hard turns would move the wheelchair away from the AP side. A solution for this could be studied and implemented to improve safety.

6.1.3 Self-localization using proprioceptive sensors

Self-localization using proprioceptive sensors, such as wheel encoders and IMUs, in addition to the exteroceptive sensors (ultrasonic sensors for contactless tethering) could further increase the accuracy in determining the powered wheelchair position next to the AP, which would be beneficial in improving the UT system's drive control algorithm by using path-planning [36]. IMU modules (accelerometers and gyroscopes), such as the MPU6050 [69], can be easily interfaced with the UT system's onboard microcontroller. IMUs output information on acceleration and rotation that tend to have a drift in baseline. The baseline drift can be corrected using a Kalman Filter [70].

6.1.4 Smartphone beacon

A smartphone has an inbuilt speaker that is capable of producing sound pulses between 20 Hz and 20 KHz. This could be used to replace the beacon developed in this project by transmitting high frequency 20 KHz pulses to identify the AP. A smartphone application can be developed to produce the pulses and enable two-way communication between the UT system and the beacon, thereby making identification a synchronous process. Modifications to the UT system would include replacing the piezoelectric ultrasonic receivers with high frequency transducers that resonate at the same frequency as the pulses (approximately 20 KHz) produced by the smartphone speaker. Although more research is required to study the effects of using high frequency sound pulses for AP identification, the use of a smartphone app would be potentially more accessible and cost effective for social-following using ultrasonic tethering.

6.1.5 Cross compatibility to other joystick emulators

Third-party joystick emulators could be used instead of the SmartChair Remote to control wheelchair movement. The UT system prototype in this research could be used as an add-on unit to any joystick emulator in the market. However, the UT system may require software modifications for joystick emulator compatibility.

6.1.6 Error detection and fail-safes

Advanced error detection algorithms can be implemented in the UT system to reduce system malfunctions. Potential errors include communication breaks between the UT system, the SmartChair Remote, or the wheelchair; or voltage spikes and sensor degradation that cause system damage or hinder system functions, which in turn could cause the wheelchair to move abruptly and dangerously. Therefore, it is necessary to implement fail-safes that disconnect the UT system and SmartChair Remote from the wheelchair. Although a fail-safe is included in the SmartChair Remote, that enables the user to manually control the powered wheelchair when necessary, the fail-safe does not take into account advanced errors related to the UT system.

References

[1] D. Kairy *et al.*, "Exploring powered wheelchair users and their caregivers' perspectives on potential intelligent power wheelchair use: A qualitative study," *Int. J. Environ. Res. Public Health*, vol. 11, no. 2, pp. 2244–2261, 2014.

[2] S. Evans, A. O. Frank, C. Neophytou, and L. de Souza, "Older adults' use of, and satisfaction with, electric powered indoor/outdoor wheelchairs," *Age Ageing*, vol. 36, no. 4, pp. 431–435, 2007.

[3] R. Evans, "The effect of electrically powered indoor / outdoor wheelchairs on occupation : a study of users ' Views," *Br. J. Occup. Ther.*, vol. 63, no. 11, pp. 547–553, 2000.

[4] R. H. Wang, A. Korotchenko, L. H. Clarke, W. Ben Mortenson, and A. Mihailidis, "Power mobility with collision avoidance for older adults: User, caregiver, and prescriber perspectives," *J. Rehabil. Res. Dev.*, vol. 50, no. 9, pp. 1287–1300, 2013.

[5] R. C. Simpson, "Smart wheelchairs: A literature review," *J. Rehabil. Res. Dev.*, vol. 42, no. 4, p. 423, 2005.

[6] L. H. V. Van Der Woude, A. J. Dallmeijer, T. W. J. Janssen, and D. Veeger, "Alternative modes of manual wheelchair ambulation: An overview," *Am. J. Phys. Med. Rehabil.*, vol. 80, no. 10, pp. 765–777, 2001.

[7] M. Shields, "Use of wheelchairs and other mobility support devices," *Stat. Canada*, vol. 15, no. 3, pp. 82–3, 2004.

[8] S. Division, "Disability in Canada," 2013.

[9] E. M. Giesbrecht, E. M. Smith, W. Ben Mortenson, and W. C. Miller, "Needs for mobility devices, home modifications and personal assistance among Canadians with disabilities," *Heal. Reports*, vol. 28, no. 8, pp. 9–15, 2017.

[10] E. M. Smith, E. M. Giesbrecht, W. B. Mortenson, and W. C. Miller, "Prevalence of wheelchair and scooter use among community-dwelling Canadians," *Phys. Ther.*, vol. 96, no. 8, pp. 1135–1142, 2016.

[11] S. Ummat and R. L. Kirby, “Nonfatal wheelchair-related accidents reported to the National Electronic Injury Surveillance System,” *Am. J. Phys. Med. Rehabil.*, vol. 73, no. 3. pp. 163–167, 1994.

[12] S. Borson and M. A. Raskind, “Clinical features and pharmacologic treatment of behavioral symptoms of Alzheimer’s disease,” *Neurology*, vol. 48, no. Issue 5, Supplement 6, p. 17S–24S, 1997.

[13] F. Masson *et al.*, “Prevalence of impairments 5 years after a head injury, and their relationship with disabilities and outcome,” *Brain Inj.*, vol. 10, no. 7, pp. 487–497, 1996.

[14] R. C. Simpson, “How many people would benefit from a smart wheelchair?,” *J. Rehabil. Res. Dev.*, vol. 45, no. 1, pp. 53–72, 2008.

[15] S. Boillée, C. Vande Velde, and D. W. W. Cleveland, “ALS: A disease of motor neurons and their nonneuronal neighbors,” *Neuron*, vol. 52, no. 1, pp. 39–59, 2006.

[16] W. Y. Chen *et al.*, “Wheelchair-related accidents: relationship with wheelchair-using behavior in active community wheelchair users,” *Arch. Phys. Med. Rehabil.*, vol. 92, no. 6, pp. 892–898, 2011.

[17] R. P. Gaal, N. Rebholtz, R. D. Hotchkiss, and P. F. Pfaelzer, “Wheelchair rider injuries : Causes and consequences for wheelchair design and selection,” *Veterans Aff. J. Rehabil. Res. Dev.*, vol. 34, no. 1, pp. 58–71, 1997.

[18] B. D. Nevitt MC, Cummings SR, Kidd S, “Risk factors for recurrent nonsyncopal falls: a prospective study.,” *Jama.*, vol. 261, pp. 2663–2668, 1989.

[19] D. Gavin-Dreschnack *et al.*, “Wheelchair-related falls: Current evidence and directions for improved quality care,” *J. Nurs. Care Qual.*, vol. 20, no. 2, pp. 119–127, 2005.

[20] G. Bourhis and Y. Agostini, “Man-machine cooperation for the control of an intelligent powered wheelchair,” *J. Intell. Robot. Syst.*, vol. 22, pp. 269–287, 1998.

[21] S. P. Levine, D. A. Bell, L. A. Jaros, R. C. Simpson, Y. Koren, and J. Borenstein, “The NavChair assistive wheelchair navigation system,” *IEEE Trans. Rehabil. Eng.*, vol. 7, no. 4, pp. 443–451, 1999.

[22] R. Simpson, E. LoPresti, S. Hayashi, I. Nourbakhsh, and D. Miller, “The Smart Wheelchair Component System,” *J. Rehabil. Res. Dev.*, vol. 41, no. 3b, p. 429, 2004.

[23] K. Wang and T. Awad, “Eightfold Technologies.” [Online]. Available: http://www.eightfoldtech.com/.

[24] M. Ben-Ari and F. Mondada, *Elements of Robotics*. 2017.

[25] H. H. Hu, P. Jia, T. Lu, and K. Yuan, “Head gesture recognition for hands-free control of an intelligent wheelchair,” *Ind. Rob.*, vol. 34, no. 1, pp. 60–68, 2007.

[26] Y. Zhang and S. C. Xu, “ROS based voice-control navigation of intelligent wheelchair,” *Appl. Mech. Mater.*, vol. 733, pp. 740–744, 2015.

[27] A. S. Kundu, O. Mazumder, P. K. Lenka, and S. Bhaumik, “Hand gesture recognition based omnidirectional wheelchair control using IMU and EMG sensors,” *J. Intell. Robot. Syst. Theory Appl.*, pp. 1–13, 2017.

[28] M. E. Lund, H. Vie Christiensen, H. A. Caltenco, E. R. Lontis, B. Bentsen, and L. N. S. Andreasen Struijk, “Inductive tongue control of powered wheelchairs,” *2010 Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBC’10*, no. Fig 1, pp. 3361–3364, 2010.

[29] T. Motokucho and N. Oda, “Vision-based human-following control using optical flow field for power assisted wheelchair,” *Int. Work. Adv. Motion Control. AMC*, pp. 266–271, 2014.

[30] J. Leaman, H. M. La, and S. Member, “A comprehensive review of smart wheelchairs: past , present and future,” *IEEE Trans. Human-Machine Systems*, vol. 47, no. 4, pp. 486–499, 2017.

[31] “Permobil Joystick Module w/Bluetooth - Permobil.” [Online]. Available: https://permobilus.com/product/permobil-joystick-module/. [Accessed: 26-Nov-2018].

[32] “Alternative Access Controls For Adult Power Wheelchairs | Numotion - Numotion.” [Online]. Available: https://www.numotion.com/products-services/adults/power/alternative-access-controls. [Accessed: 26-Nov-2018].

[33] “Sip/Puff Switch.” [Online]. Available: http://www.orin.com/access/sip_puff/. [Accessed: 26-Nov-2018].

[34] K. Arai and R. Mardiyanto, “Eyes based eletric wheel chair control system -i (eye) can control electric wheel chair,” *IJACSA) Int. J. Adv. Comput. Sci. Appl.*, vol. 2, no. 12, pp. 98–105, 2011.

[35] S. Desai, S. S. Mantha, and V. M. Phalle, “Advances in smart wheelchair technology,” *2017 Int. Conf. Nascent Technol. Eng. ICNTE 2017 - Proc.*, no. January, 2017.

[36] H. R. Everett, “Sensors for Mobile Robots: Theory and Application,” *CRC Press*, vol. 12, no. 6, p. 922, 1996.

[37] J. Illade-Quinteiro, V. M. Brea, P. López, D. Cabello, and G. Doménech-Asensi, “Distance measurement error in time-of-flight sensors due to shot noise,” *Sensors (Switzerland)*, vol. 15, no. 3, pp. 4624–4642, 2015.

[38] D. Marioli, C. Narduzzi, C. Offelli, D. Petri, E. Sardini, and A. Taroni, “Digital time-of-flight measurement for ultrasonic sensors,” *IEEE Trans. Instrum. Meas.*, vol. 41, no. 1, pp. 93–97, 1992.

[39] M. Alshraideh, B. A. Mahafzah, S. Al-Sharaeh, and Z. M. Hawamdeh, “A robotic intelligent wheelchair system based on obstacle avoidance and navigation functions,” *J. Exp. Theor. Artif. Intell.*, vol. 27, no. 4, pp. 471–482, 2015.

[40] C. Cadena *et al.*, “Past, present, and future of simultaneous localization and mapping: towards the robust-perception age,” *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, 2016.

[41] A. Ravankar, A. A. Ravankar, Y. Kobayashi, Y. Hoshino, and C. C. Peng, “Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges,” *Sensors (Switzerland)*, vol. 18, no. 9, pp. 1–30, 2018.

[42] F. A. A. Cheein, C. De La Cruz, T. F. Bastos, and R. Carelli, “Slam-based cross-a-door solution approach for a robotic wheelchair,” *Int. J. Adv. Robot. Syst.*, vol. 7, no. 2, pp. 155–164, 2010.

[43] J. U. Ch, I. Youn, K. Choi, and Y. J. Lee, “Human-following robot using tether steering,” *Int. J. Precis. Eng. Manuf.*, vol. 12, no. 5, pp. 899–906, 2011.

[44] N. Kawarazaki, L. T. Kuwae, and T. Yoshidome, “Development of human following mobile robot system using laser range scanner,” *Procedia Comput. Sci.*, vol. 76, no. Iris, pp. 455–460, 2015.

[45] Quoc Khanh Dang and Young-Soo Suh, “Human-following robot using infrared camera,” pp. 1054–1058, 2011.

[46] S. Na, H. S. Ahn, Y. C. Lee, and W. Yu, “Navi-guider: An intuitive guiding system for the mobile robot,” *Proc. - IEEE Int. Work. Robot Hum. Interact. Commun.*, pp. 228–233, 2007.

[47] Y. Kobayashi, R. Suzuki, and Y. Kuno, “Robotic wheelchair with omni-directional vision for moving alongside a caregiver,” in *IECON Proceedings (Industrial Electronics Conference)*, 2012, pp. 4177–4182.

[48] J. Zhang, J. Wang, and W. Chen, “A control system of driver assistance and human following for smart wheelchair,” *2014 IEEE Int. Conf. Robot. Biomimetics, IEEE ROBIO 2014*, pp. 1927–1932, 2014.

[49] K. Miyazaki, M. Hashimoto, M. Shimada, and K. Takahashi, “Guide following control using laser range sensor for a smart wheelchair,” *ICROS-SICE Int. Jt. Conf.*, pp. 4613–4616, 2009.

[50] “Kobuki – Yujinrobot | Robot Vacuums.” [Online]. Available: http://en.yujinrobot.com/archives/portfolio-items/kobuki?portfolioCats=85%2C64. [Accessed: 12-Nov-2018].

[51] B.-F. Wu, C.-L. Jen, T.-Y. Tsou, W.-F. Li, and P.-Y. Tseng, “Accompanist detection and following for wheelchair robots with fuzzy controller,” *Adv. Mechatron. Syst. (ICAMechS), 2012 Int. Conf.*, pp. 638–643, 2012.

[52] E. T. Hall, “A System for the Notation of Proxemic Behavior,” *Am. Anthropol.*, vol. 65, no. 5, pp. 1003–1026, 1963.

[53] T. Pingali, E. Lemaire, and N. Baddour, “Ultrasonic tethering to enable side-by-side following for powered wheelchairs,” *Sensors*, vol. 19, no. 1, p. 109, 2018.

[54] T. Mohammad, “Using ultrasonic and infrared sensors for distance measurement,” *Eng. Technol.*, pp. 293–298, 2009.

[55] J. Kim, B. L. Grisso, J. K. Kim, D. S. Ha, and D. J. Inman, “Electrical modeling of piezoelectric ceramics for analysis and evaluation of sensory systems,” *IEEE Sensors Appl. Symp.*, pp. 122–127, 2008.

[56] MathWorks, “Simulink Documentation,” 2018. [Online]. Available: https://www.mathworks.com/help/simulink/. [Accessed: 14-Oct-2018].

[57] L. Fehr, W. E. Langbein, and S. B. Skaar, “Adequacy of power wheelchair control interfaces for persons with severe disabilities: a clinical survey.,” *J. Rehabil. Res. Dev.*, vol. 37, no. 3, pp. 353–60, 2000.

[58] K. G. Panda, D. Agrawal, A. Nshimiyimana, and A. Hossain, “Effects of environment on accuracy of ultrasonic sensor operates in millimetre range,” *Perspect. Sci.*, vol. 8, pp. 574–576, 2016.

[59] MaxBotix, “LV-MaxSonar ® -EZ™ Series,” pp. 1–12, 2015.

[60] S. Ogiso, T. Kawagishi, K. Mizutani, N. Wakatsuki, and K. Zempo, “Self-localization method for mobile robot using acoustic beacons,” *ROBOMECH J.*, vol. 2, no. 1, 2015.

[61] M. Drumheller, “Mobile robot localization using sonar,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 2, p. 1, 1987.

[62] F. Donoso-Aguirre, J. P. Bustos-Salas, M. Torres-Torriti, and A. Guesalaga, “Mobile robot localization using the Hausdorff distance,” *Robotica*, vol. 26, no. 2, pp. 129–141, 2008.

[63] J. Borenstein and Y. Koren, “Obstacle avoidance with ultrasonic sensors,” *IEEE J. Robot. Autom.*, vol. 4, no. 2, pp. 213–218, 1988.

[64] R. Knoblauch, M. Pietrucha, and M. Nitzburg, “Field studies of pedestrian walking speed and start-up time,” *Transp. Res. Rec. J. Transp. Res. Board*, vol. 1538, pp. 27–38, Jan. 1996.

[65] D. A. Winter, “Human balance and posture control during standing and walking,” *Gait and Posture*, vol. 3, no. 4. pp. 193–214, 1995.

[66] D. González, J. Pérez, V. Milanés, and F. Nashashibi, “A review of motion planning techniques for automated vehicles,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1135–1145, 2016.

[67] Y. Ting and H.-C. Jar, “Determination of location and path planning algorithms for industrial robots,” *Control*, vol. 3, no. December, pp. 379–392, 2006.

[68] B. Gersdorf, “A castor wheel controller for differential drive,” pp. 174–179.

[69] I. Inc., “MPU-6000 and MPU-6050 Product Specification,” vol. 1, no. 408. 2013.

[70] C. K. Chui and G. Chen, *Kalman filtering with real-time applications*, Fifth edition, 2005.

[71] Recom, “R-78E-0.5 DC DC Converter.”

[72] Atmel, “Atmel ATmega DataSheet,” 2014.

[73] “MAX232x Dual EIA-232 Drivers/Receivers,” 2014.

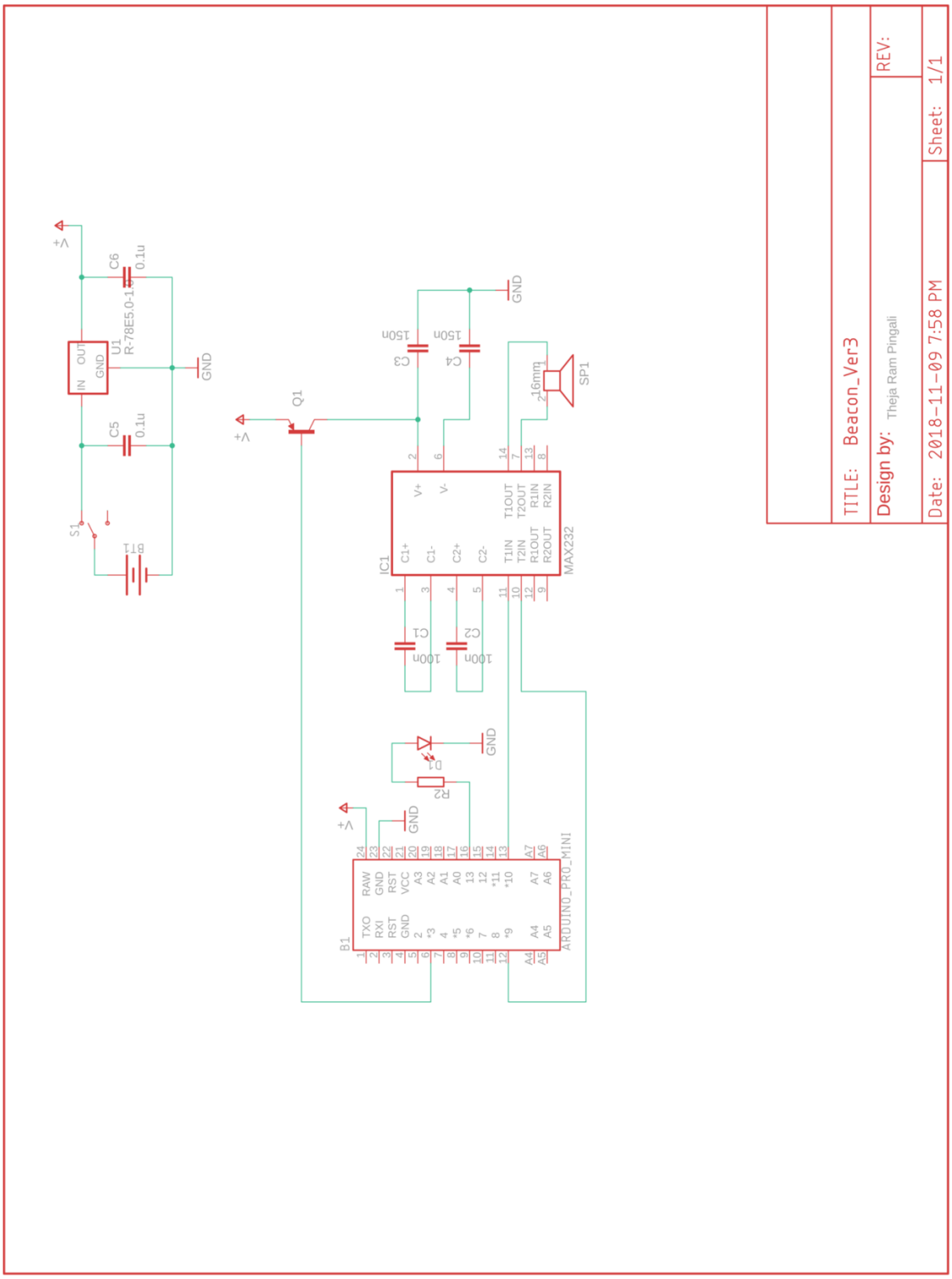
[74] Jo-Ann Hogan, “Getting started with Fusion 360,” 2014.

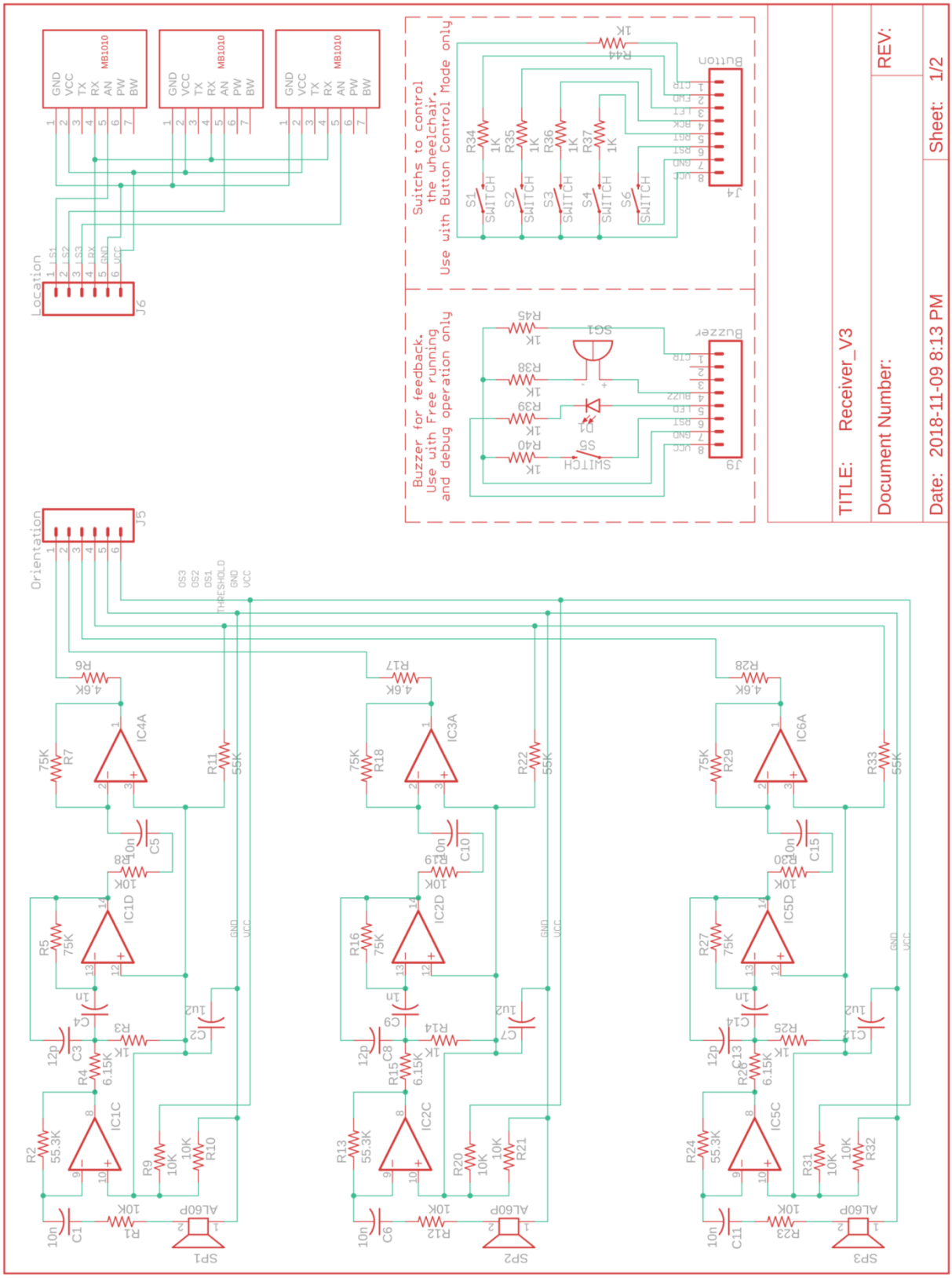
Appendix A

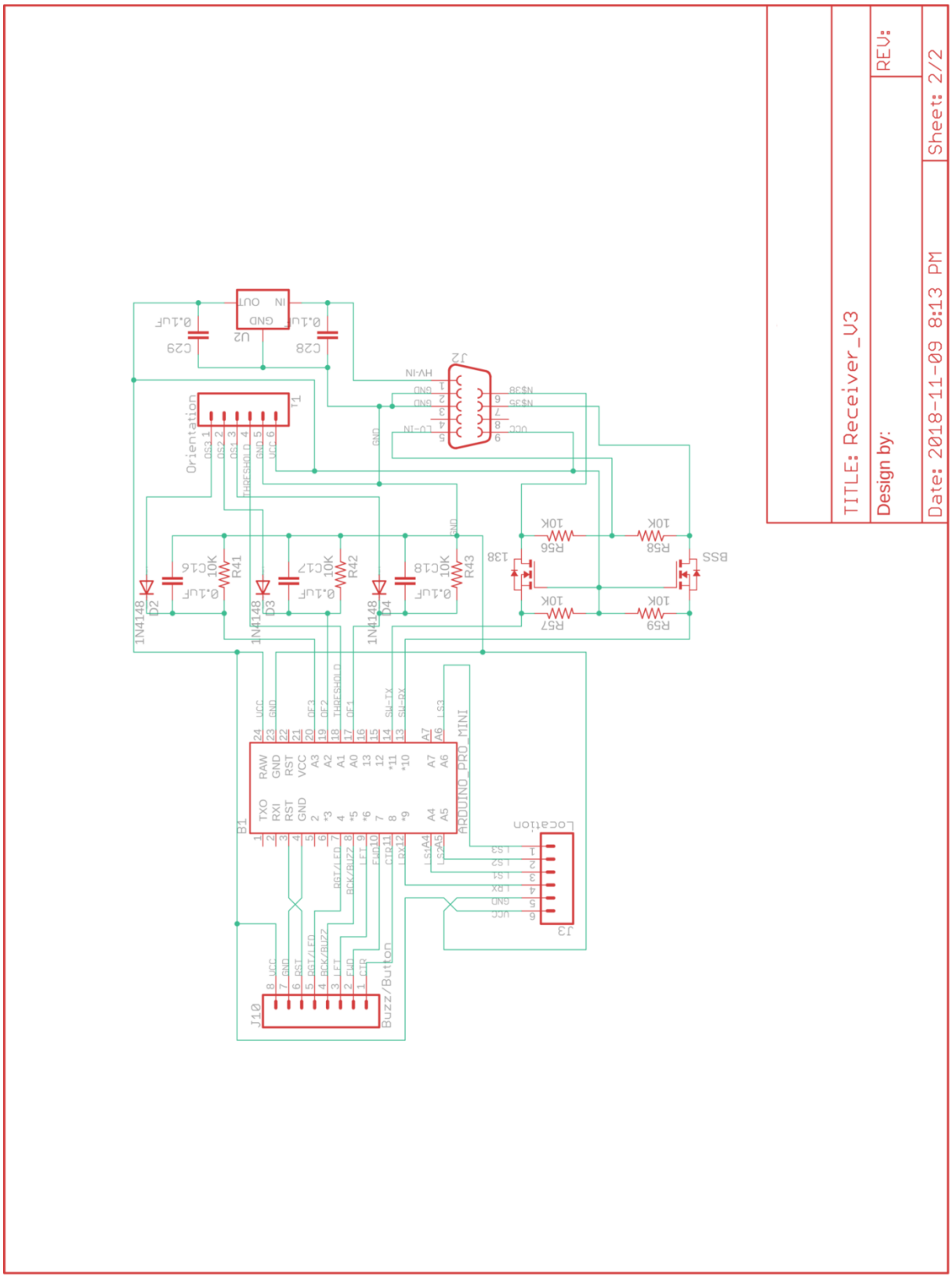
This section consists of the ultrasonic tethering circuit schematics. The overall circuit included three sub-schematics, a beacon, receiver, and microcontroller processing schematic. The beacon schematic consisted of a power supply/voltage conditioner circuit using a commercially available 9 V battery and a R-78E-0.5 voltage switching regulator that had a higher input to output efficiency compared to commonly used voltage regulators [71]. The beacon schematic also consisted of an Atmel-based microcontroller (ATmega328) to produce two 40 KHz pulses of 180-degree phase shift with an amplitude of 0 – 5V using an inbuilt 16-bit hardware timer (Timer 1) [72]. The circuit was designed to produce a 40 KHz pulse with maximized peak-to-peak voltage to drive a piezoelectric ultrasonic transducer. The pulses were fed into a MAX232 IC, which is a transistor-transistor logic (TTL) to RS-232 (and vice versa) converter. TTL is a 5 V logic and TIA/EIA232 is a 20 V logic, that converts the 5 V input signal to 20 V. [73].

A Bipolar Junction Transistor (BJT) was used to switch the voltage driver circuit on or off, for implementing addressing. The receiver circuit consisted of three piezoelectric ultrasonic sensor amplification and filtering circuits, two ports for the MaxBotix MB1010 ping sensor (one backup port and two live ports) and one port for the user interface (button circuit to control wheelchair direction and buzzer circuit for accompanying person feedback). All signals were transmitted to the microcontroller board for further processing.

The microcontroller circuit consisted of a power supply/voltage conditioning circuit that converted the 24 V supply from the wheelchair battery to a manageable 5 V supply. The microcontroller circuit also consisted of serial communication circuits (with a level shifter) to transmit powered wheelchair direction commands to the SmartChair remote by Eightfold Technologies. Serial communication between the SmartChair remote and the ultrasonic tethering system occurred via a DB9 connector and a multi-strand six core shielded cable. The circuits were designed based on the simulation results presented in Chapter 3. However, changes were made to the final amplification and filtering circuits while prototype testing. The optimized passive component parameter values, such as resistors and capacitors, from the simulations resulted in a required gain for the amplifiers and cut-off frequency for the low-pass filters. These gain and low-pass filter cut-off values were similar from the simulation to the prototype. Second stage filters were implemented digitally on the ATmega328 microcontroller.


TITLE: Beacon_Ver3
Design by: Theja Ram Pingali
REV:
Date: 2018-11-09 7:58 PM
Sheet: 1/1
MAX232
ARDUINO_PRO_MINI
16mm
R-78E5.0-1.0
GND
0.1u
150n
100n


Switchs to control
the wheelchair.
Use with Button Control Mode only
Buzzer for feedback.
Use with Free running
and debug operation only
TITLE: Receiver_V3
Document Number:
REV:
Date: 2018-11-09 8:13 PM
Sheet: 1/2
Location
Orientation
Button
Buzzer
GND
VCC
TX
RX
AN
PW
BW
MB1010
OS3
OS2
OS1
THRESHOLD
IC1C
IC1D
IC2C
IC2D
IC4A
IC3A
IC5C
IC5D
IC6A
AL60P


TITLE: Receiver_U3
Design by:
REV:
Date: 2018-11-09 8:13 PM
Sheet: 2/2
Orientation
Location
Buzz/Button
ARDUINO_PRO_MINI
1N4148
10K
0.1uF

1 Bill of materials: Beacon circuit (components, descriptions and parameters)

Component	Part Number	Description (Package)	Values (Units)
U1	R-78E5.0-0.5	DC-DC converter (3-SIP[1])	5 V, 3 W
B1	DEV-11113	Atmega328 Arduino	5 V, 16 MHz
IC1	MAX232	EIA-232 driver (DIP[2])	<6 V
Q1	2N3906-AP	PNP transistor (TO-92[3])	40 V, 0.2 A
SP1	MSO-AT1640H10R	Piezoelectric transducer	40 KHz, ∅ 10mm
BT1	ID1604	Battery	9 V
S1	CKN10397-ND	Switch slide (SPDT[4])	200 mA, 30 V
C1, C2, C5, C6	K104K15X7RF5TL2	Ceramic disk capacitor (TH[5])	0.1 uF ± 10%, 50 V
C3, C4	K154Z15Y5VF53L2	Ceramic disk capacitor (TH)	0.15 uF, 50V
R1, R2	CF14JT100R	Resistor (TH)	100 Ω ± 5%, 0.25 W
D1	HLMP-1301	LED diffused (red) (TH)	626 nm, 1.9 V

2 Bill of materials: Receiver circuit (components, descriptions and parameters)

Component	Part Number	Description (Package)	Values (Units)
U2	R-78E5.0-1.0	DC-DC converter (3-SIP)	5 V, 5 W
B1	DEV-11113	Atmega328 Arduino	5 V, 16 MHz
IC1, IC2, IC3	LM324AN	General Purpose op-amp (DIP)	1.2MHZ
MOD1, MOD2, MOD3	MB1010	LV-MaxSonar – EZ	~5 V, 42 KHz
SG1	TMB12A05	Buzzer (TH)	<7 V, <30 mA, >85 dB
SP1, SP2, SP3	MSO-AT1640H10R	Piezoelectric transducer	40 KHz, ∅ 10mm
S1, S2, S3, S4, S5, S6	B3F-1002S	Tactile Switch (SPST-TH)	24 V, 0.05 A
Q1	BSS138	N-channel MOSFET (SOT-23)	50 V, 220 mA, 360 mW
D2, D3, D4	1N4148-T	General Purpose Diode (DO35)	75 V, 150 mA
C1, C6, C11	K103K15X7RF5TL2	Ceramic disk capacitor (TH)	10 nF ± 10%, 50 V

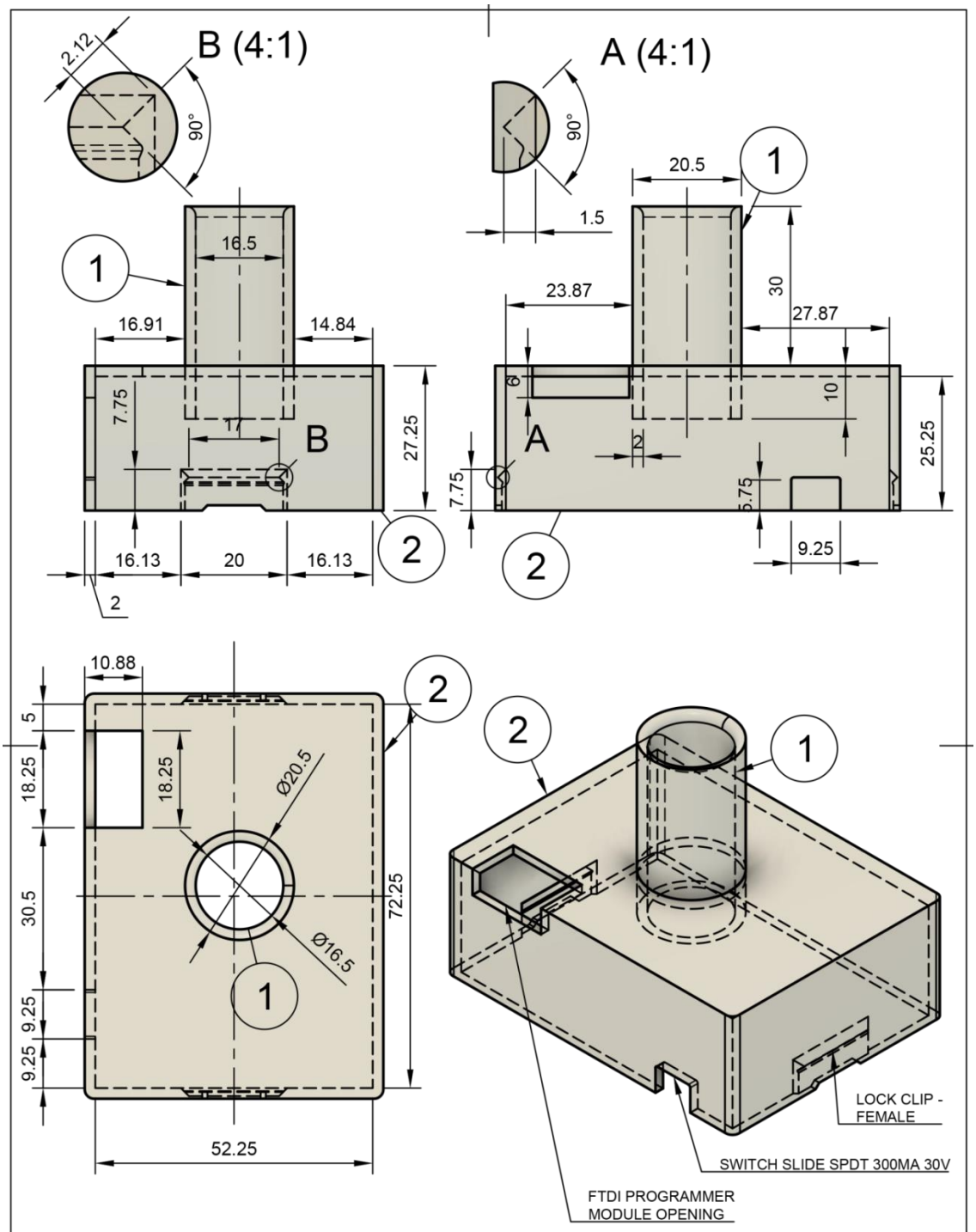
[1] SIP – System in package
[2] DIP – Dual in-line package
[3] TO-92 - Transistor Outline Package, Case Style 92
[4] SPDT – Single pole dual throw
[5] TH – Through hole package

C2, C7, C12	K105Z20Y5VF5TL2	Ceramic disk capacitor (TH)	1 uF, 50 V
C3, C8, C13	K120J15C0GF5TL2	Ceramic disk capacitor (TH)	12 pF ± 5%, 50V
C4, C9, C14	K102K15X7RF5TL2	Ceramic disk capacitor (TH)	1 nF ± 10%, 50V
C5, C10, C15	K103K15X7RF5TL2	Ceramic disk capacitor (TH)	10 nF ± 10%, 50 V
C16, C17, C18	K104K15X7RF5TL2	Ceramic disk capacitor (TH)	0.1 uF ± 10%, 50 V
C28, C29	K104K15X7RF5TL2	Ceramic disk capacitor (TH)	0.1 uF ± 10%, 50 V
R1, R12, R23	CF14JT10K0	Resistor (TH)	10 KΩ ± 5%, 0.25 W
R2, R13, R24	RN55D5502FB14	Resistor (TH)	55.3 KΩ ± 1%, 0.125 W
R3, R14, R25	CF14JT1K00	Resistor (TH)	1 KΩ ± 5%, 0.25 W
R4, R15, R26	CF14JT4K70	Resistor (TH)	6.15 KΩ ± 5%, 0.25 W
R5, R16, R27	MRS25000C7502FRP00	Resistor (TH)	75 KΩ ± 1%, 0.6 W
R6, R17, R28	CF14JT4K70	Resistor (TH)	4.7 KΩ ± 5%, 0.25 W
R7, R18, R29	MRS25000C7502FRP00	Resistor (TH)	75 KΩ ± 1%, 0.6 W
R8, R19, R30	CF14JT10K0	Resistor (TH)	10 KΩ ± 5%, 0.25 W
R9, R20, R31	CF14JT10K0	Resistor (TH)	10 KΩ ± 5%, 0.25 W
R10, R21, R32	CF14JT10K0	Resistor (TH)	10 KΩ ± 5%, 0.25 W
R11, R22, R33	RN55D5502FB14	Resistor (TH)	55 KΩ ± 1%, 0.125 W
R34, R35, R36, R37	CF14JT1K00	Resistor (TH)	1 KΩ ± 5%, 0.25 W
R38, R39, R40, R45	CF14JT1K00	Resistor (TH)	1 KΩ ± 5%, 0.25 W
R41, R42, R43	CF14JT10K0	Resistor (TH)	10 KΩ ± 5%, 0.25 W
R56, R57, R58, R59	CF14JT10K0	Resistor (TH)	10 KΩ ± 5%, 0.25 W
J1, J3	D01-9923246	Header Connector - male (SIP)	6 pin connectors
J2	A-DF 09 A/KG-T2S	D-Sub connector	9 position D-sub (DB9)
J10	D01-9923246	Header Connector - male (SIP)	6 pin connectors
J4, J9	CN-PH-F140S	Header Connector - female (SIP)	8 pin connectors
J5, J6	CN-PH-F140S	Header Connector - female (SIP)	6 pin connectors
D1	HLMP-1301	LED diffused (red) (TH)	626 nm, 1.9 V

Appendix B

This section consists of 2-D computer aided designs (CAD) and images for the beacon and the receiver. The beacon was mounted on the AP, on the side towards the powered wheelchair using a belt hook. The belt hook consisted of two openings for a Velcro strip that can be wound around a waist belt, securing the beacon. The beacon was attached to the belt hook using a pivot hinge and fastened with five 3mm nuts and bolts. The beacon consisted of a signal guider, top cover, base with PCB riser, belt hook, pivot hinge, and ∅3mm hex nuts and bolts. The main PCB was snugly fit inside the top cover and the base, with the piezoelectric transducer fit inside the signal guider. The top cover and base were held together using snap fit clips. The signal guider was designed to guide the ultrasonic pulses towards the wheelchair receiver, by reducing the throw angle from ±15º to ±8º. This allowed finer signal comparisons between each receiver on the wheelchair. The receiver consisted of a top cover, base cover with PCB riser, sensor mount (wing), slide for easy receiver mounting, pivot hinge, U-clamp, and ∅3mm hex nuts and bolts. The top cover and base were held together using snap fit clips. The sensor wing was attached to the top cover using a friction slide. All sensors were mounted on the wing using 3M double sided sticky tape. The pivot hinges could be used to vertically adjust the angles, to maintain an approximate 'line-of-sight'. All 2-D CAD drawings were 3-D designed on Autodesk Fusion 360 version 2.0.4860 (Education License) [74]. The designed components were 3-D printed using a Makerbot Ultimaker 2 Extended 3-D printer and sliced on Ultimaker Cura. The table below describes the parameters and configuration used for 3-D printing the system.

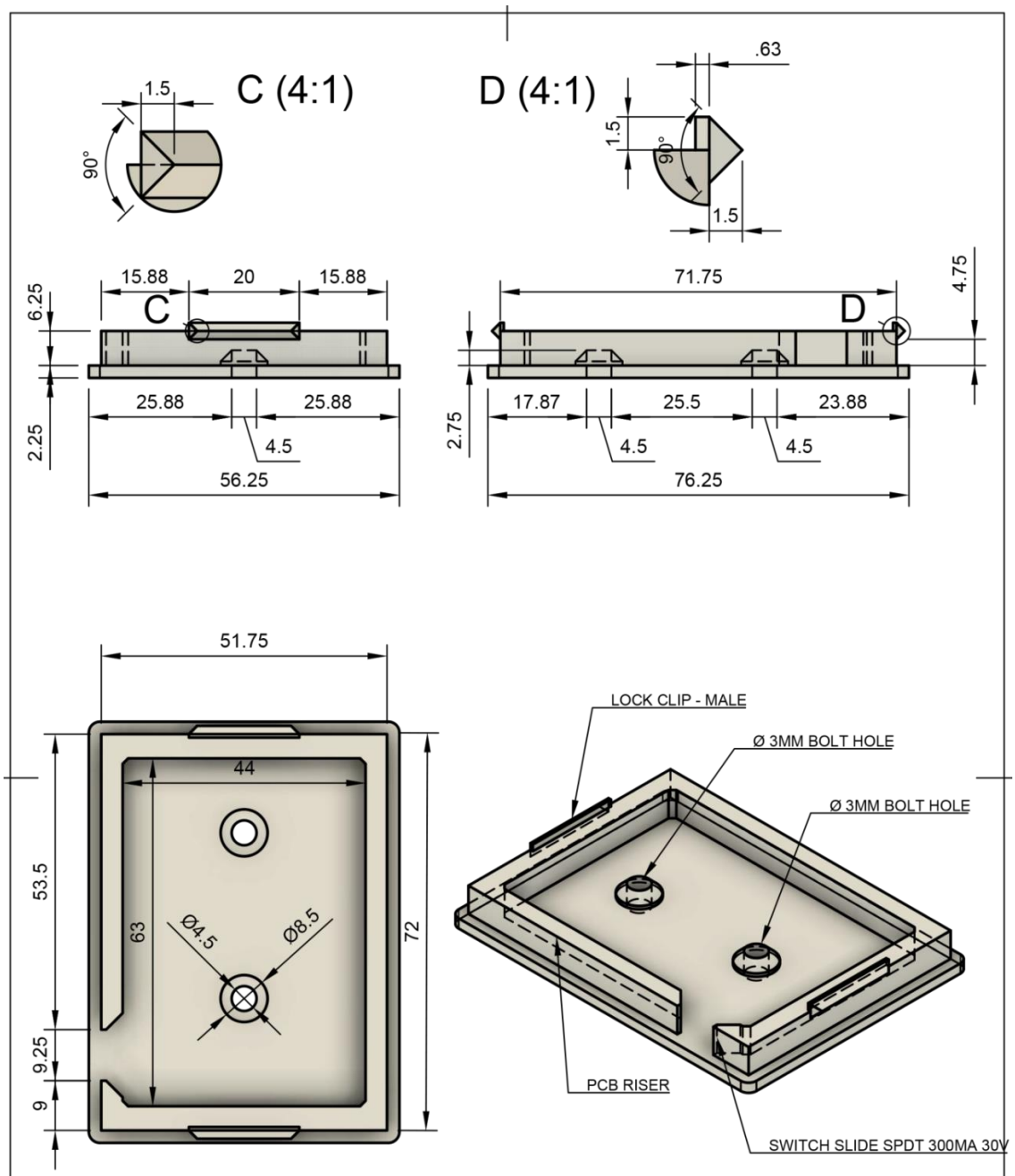
3-D Printer description	Configuration
Nozzle size	0.4 mm
Printer speed	50 mm/sec
Layer height	0.2 mm
Wall thickness	0.4 mm
Infill density	50%
Infill pattern	Triangles



2	Signal Guider - Cylinder	PLA Plastic: 0.4mm Nozzle
1	Top cover	PLA Plastic: 0.4mm nozzle
Sl No	**Part**	**Properties**

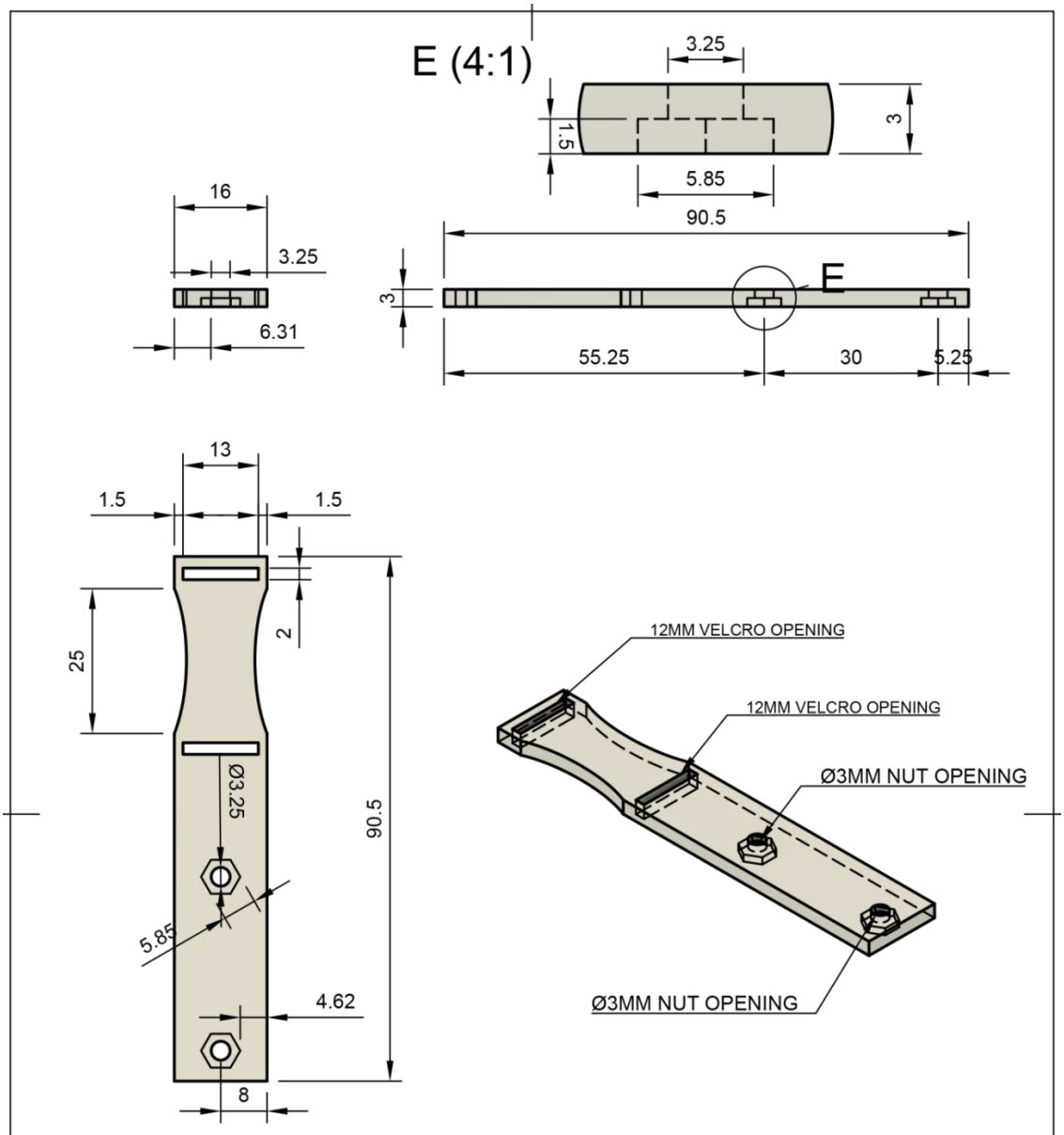
Beacon_Top

Created by	Title	Scale	Units
Theja Ram Pingali	Beacon Top w/ signal guider	1:1	mm
		Rev. 1 / Date of issue / Sheet 1	



1	Base	PLA Plastic: 0.4mm nozzle
Sl No	**Part**	**Properties**
Beacon_Bottom		

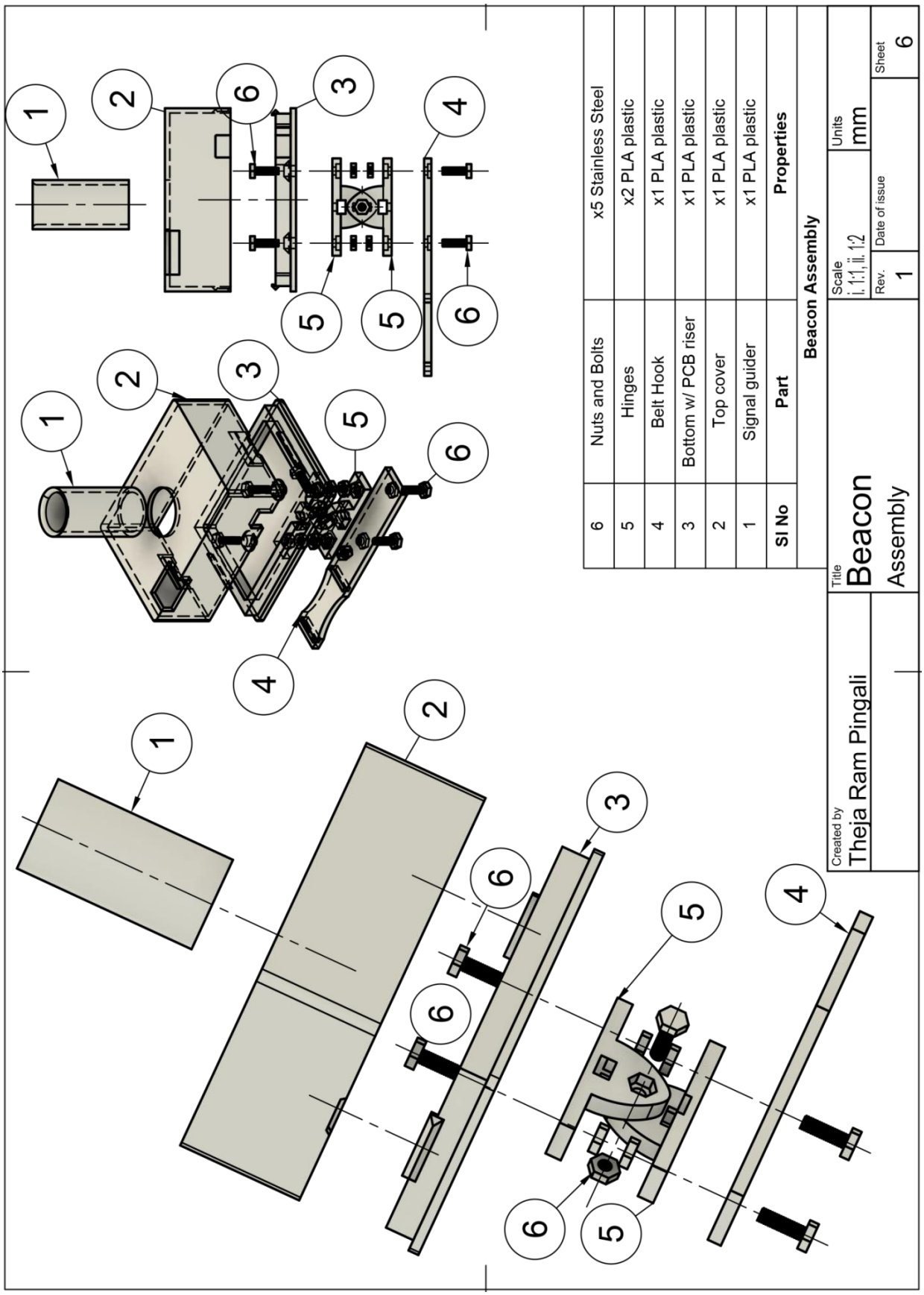
Created by: Theja Ram Pingali	Title: Beacon Bottom w/ PCB riser	Scale: 1:1	Units: mm	
		Rev.: 1	Date of issue:	Sheet: 2

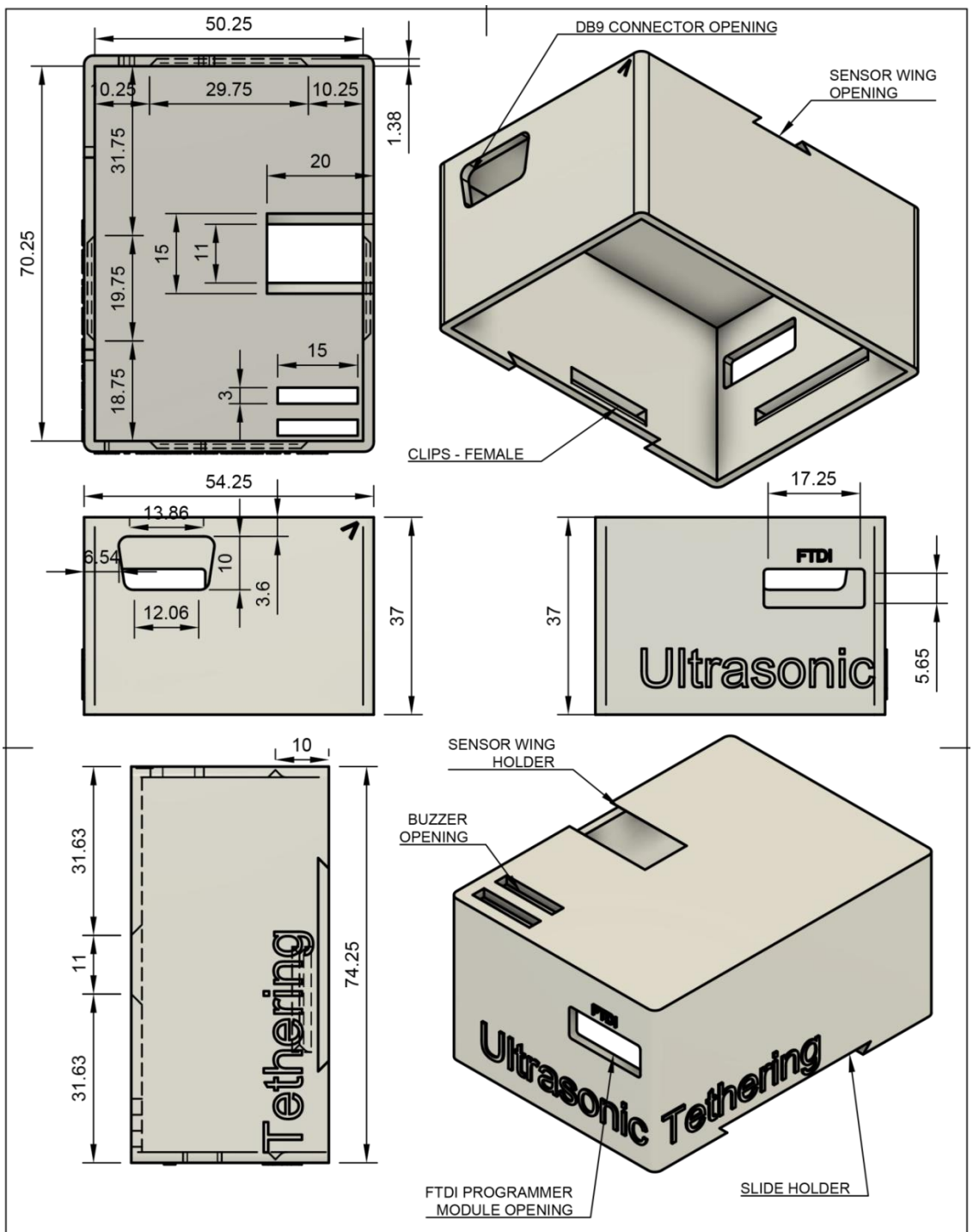


1	Belt Hook	PLA Plastic: 0.4mm nozzle
Sl No	**Part**	**Properties**

Beacon_Belt Hook

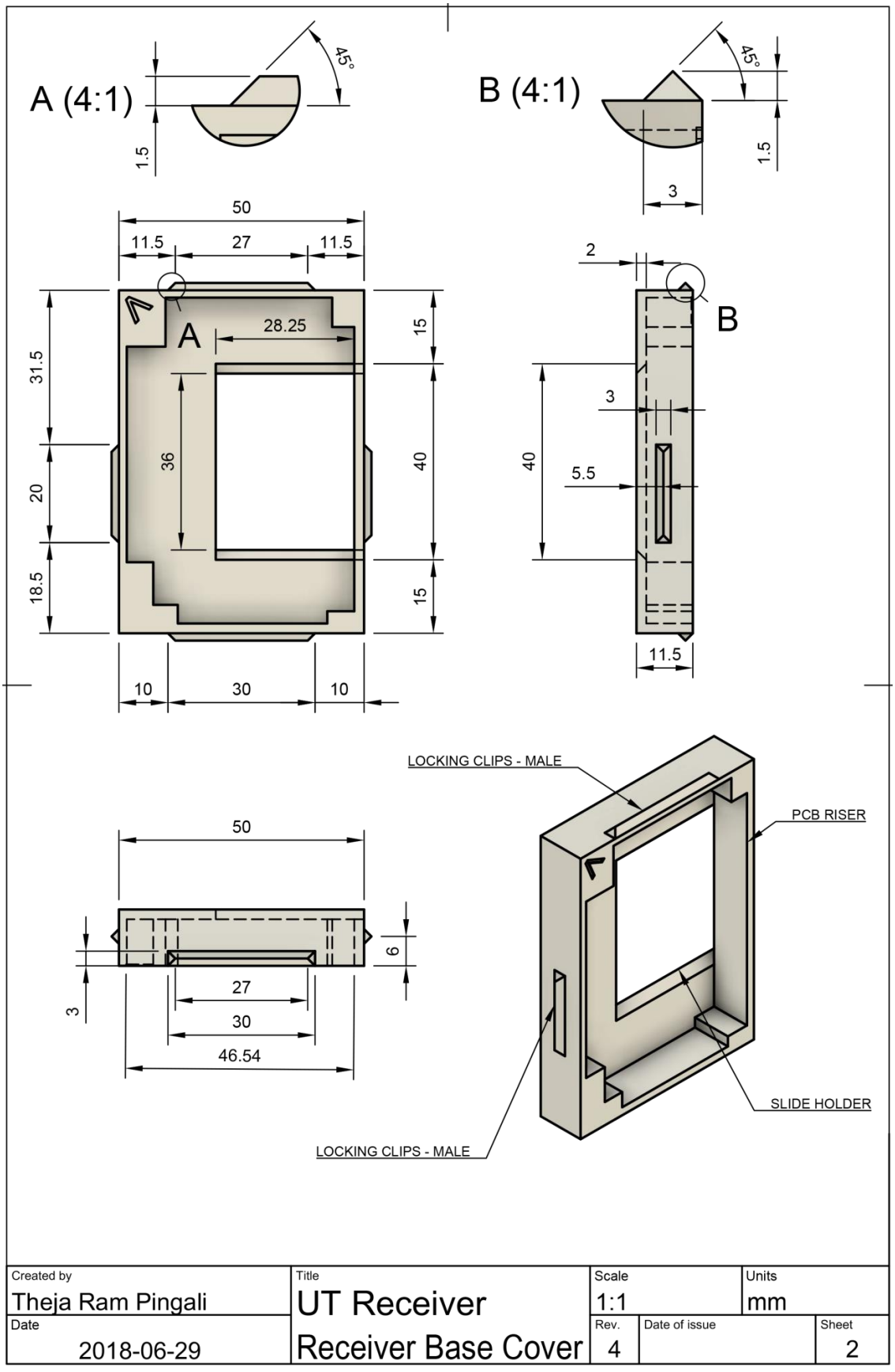
Created by: Theja Ram Pingali	Title: Beacon Belt Hook	Scale: 1:1	Units: mm	
		Rev.: 1	Date of issue:	Sheet: 3


1
2
6
3
5
5
4
6
1
2
3
5
6
4
1
2
6
6
3
5
6
5
4
6 Nuts and Bolts x5 Stainless Steel
5 Hinges x2 PLA plastic
4 Belt Hook x1 PLA plastic
3 Bottom w/ PCB riser x1 PLA plastic
2 Top cover x1 PLA plastic
1 Signal guider x1 PLA plastic
Sl No Part Properties
Beacon Assembly
Created by
Theja Ram Pingali
Title
Beacon
Assembly
Scale
i. 1:1, ii. 1:2
Rev.
1
Date of issue
Units
mm
Sheet
6

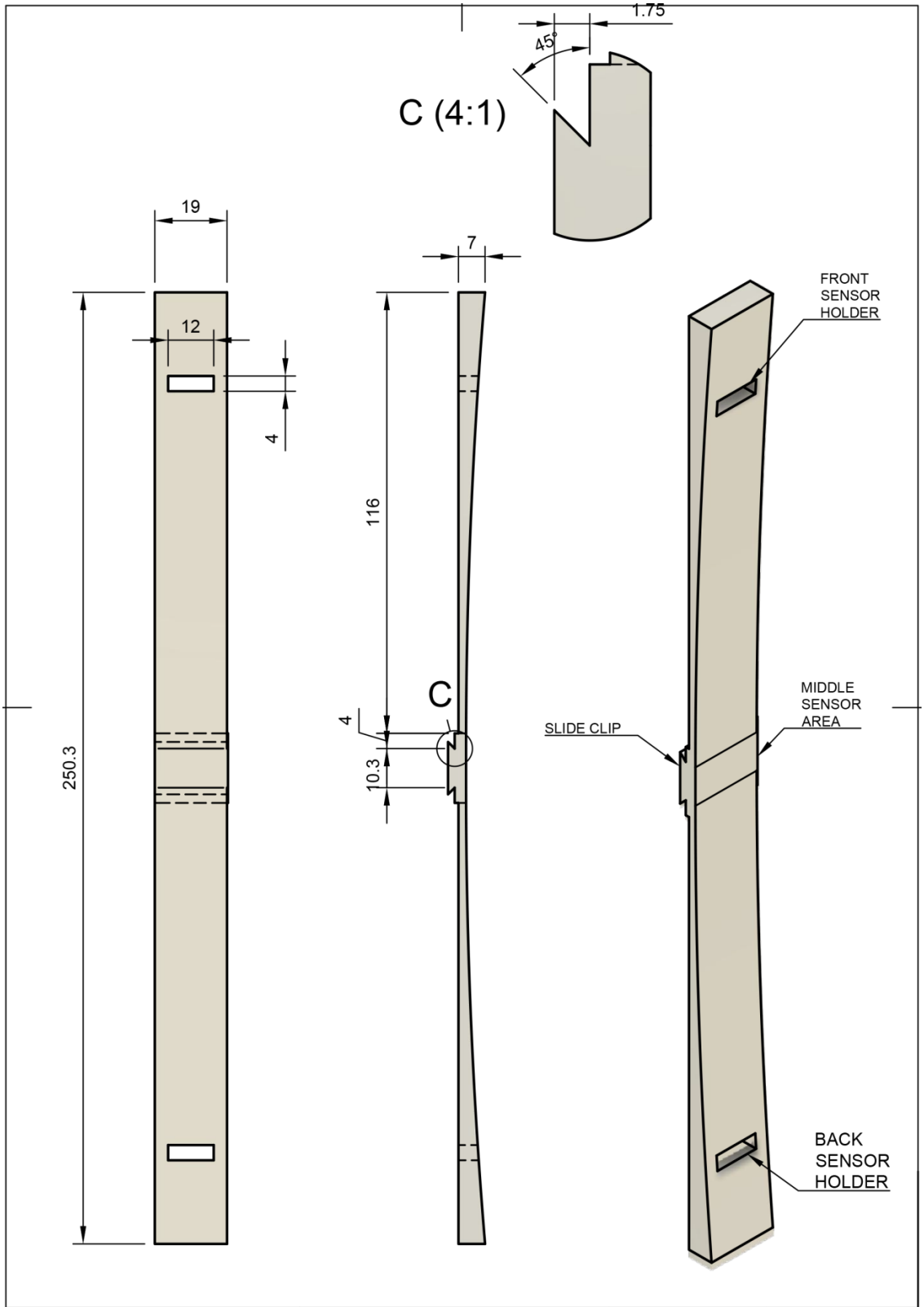


1	Receiver Cover - Top	Shell Thickness - 2mm, PLA Plastic
Sl No	**Part**	**Properties**

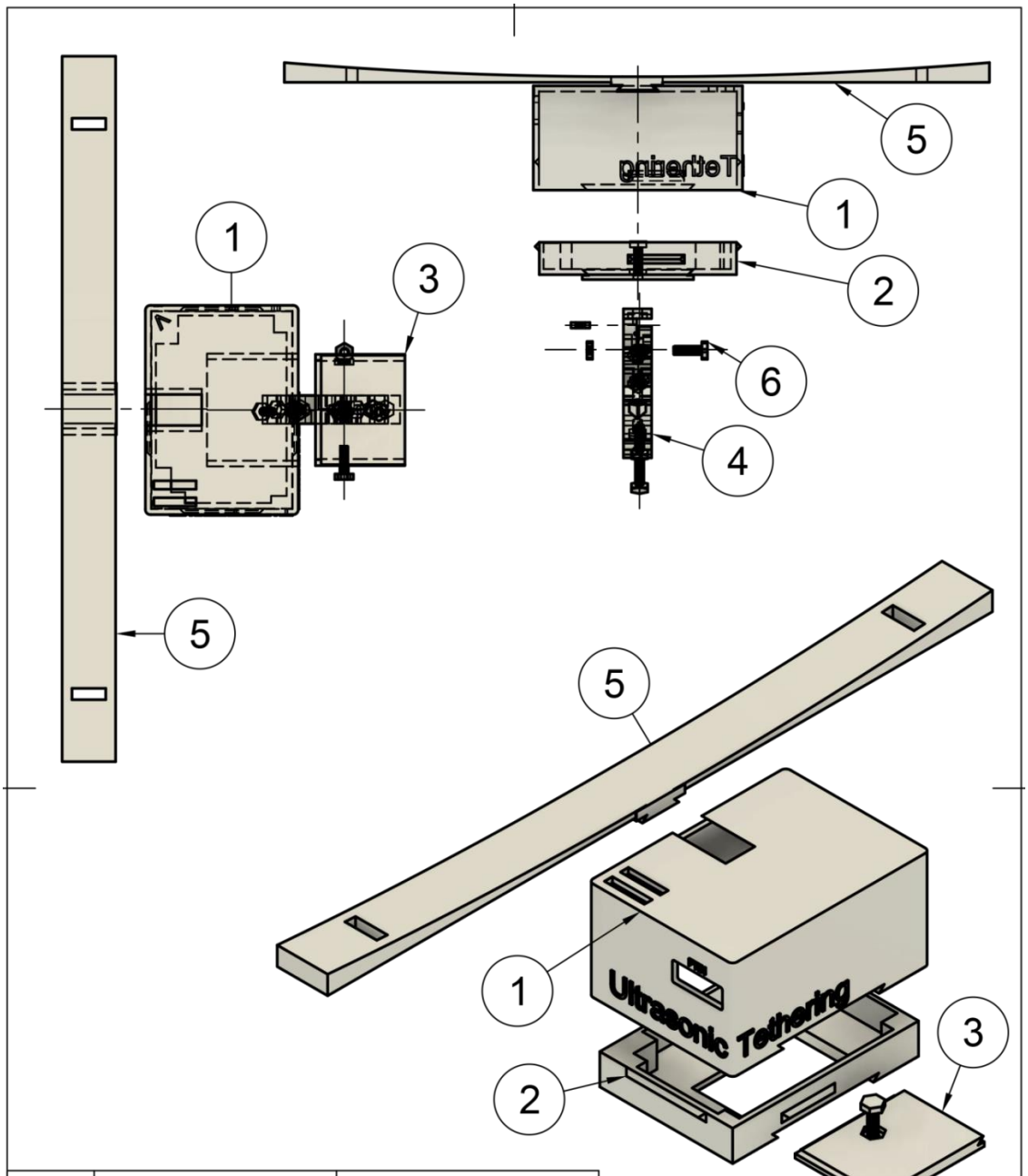
Created by	Title	Scale	Units
Theja Ram Pingali	UT Receiver Receiver Top Cover	1:1	mm
Date: 2018-06-29		Rev. 4 / Date of issue / Sheet 1	



Created by	Title	Scale	Units
Theja Ram Pingali	UT Receiver	1:1	mm
Date		Rev. / Date of issue	Sheet
2018-06-29	Receiver Base Cover	4	2



Created by	Title	Scale	Units
Theja Ram Pingali	UT Receiver Sensor Holder	0.75:1	mm
Date: 2018-06-29		Rev.: 4	Date of issue:	Sheet: 3



6	Nuts and Bolts	3MM Stainless Steel
5	Sensor Holder (wing)	PLA Plastic
4	Hinges	PLA Plastic
3	Sensor Module Holder	PLA Plastic
2	UT Receiver Base	PLA Plastic
1	UT Receiver Top Cover	PLA Plastic
Sl No	**Part**	**Properties**

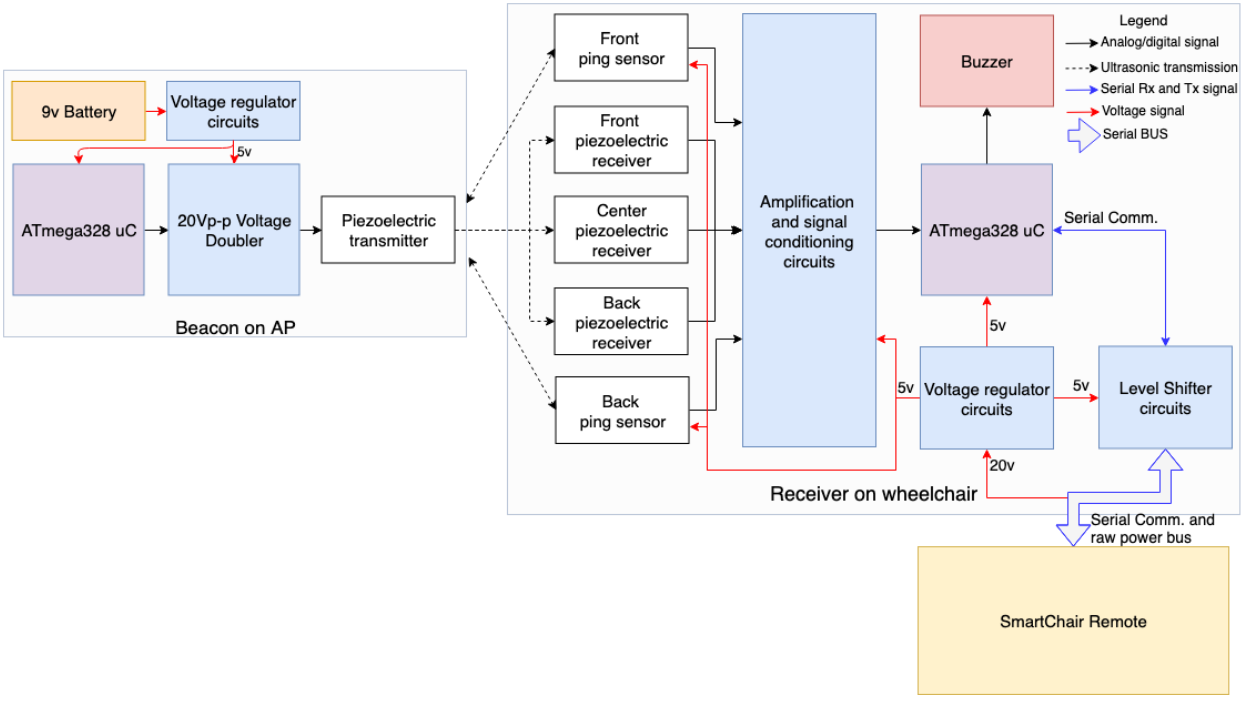
Created by: Theja Ram Pingali	Title: UT Receiver Assembly	Scale: 1:2, 1:1.5	Units: mm
Date: 2018-06-29		Rev.: 4 / Date of issue:	Sheet: 1

Appendix C

The ultrasonic tethering system discussed in this thesis combined multiple processes, from AP identification to controlling the wheelchair direction by producing appropriate wheelchair direction commands. This section describes the overall processes to achieve contactless tethering between the AP and the powered wheelchair using ultrasonic sensors. The UT system hardware block diagram is divided into two parts, the beacon on the AP and the receiver on the wheelchair, connected to the SmartChair Remote.

The overall flow diagram can be divided into four parts: sensor calibration, identification, determining the AP pose and velocity (tether distance and tether angle), and producing appropriate wheelchair direction and speed commands. Variables are defined in subsection 2.

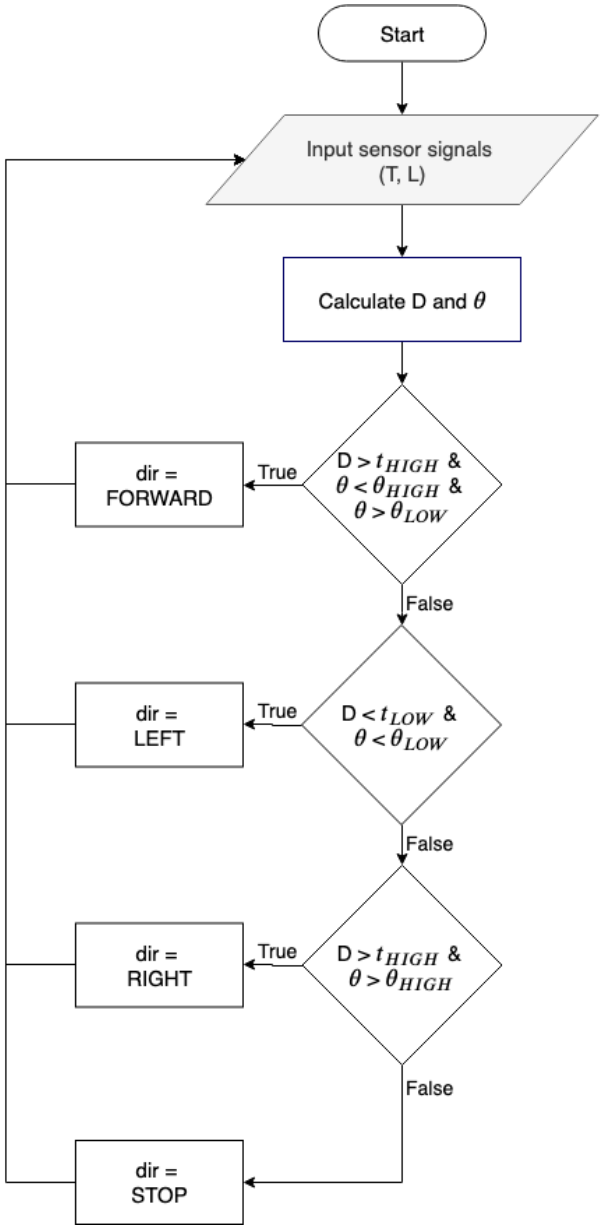
1 Ultrasonic tethering system hardware block diagram



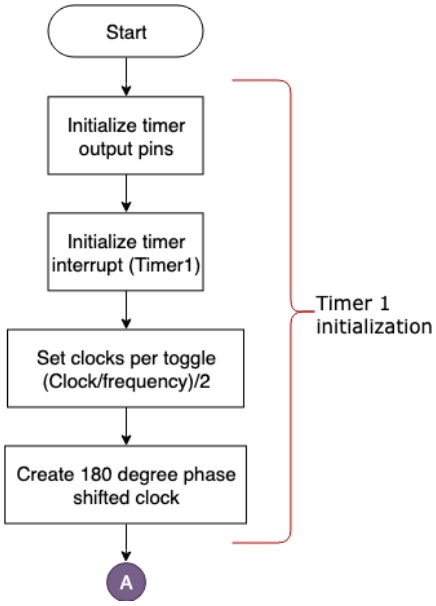
2 Variables and definitions

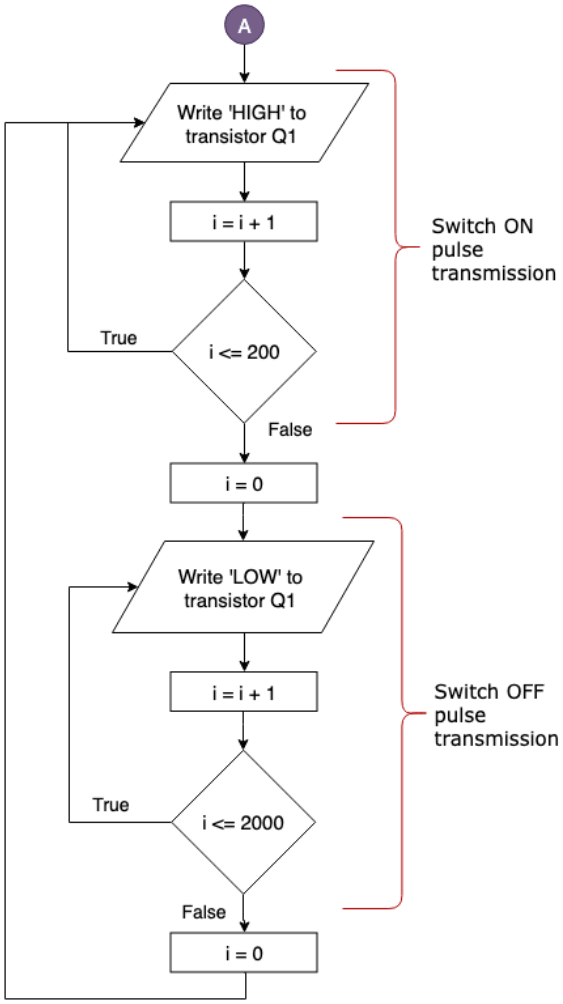
Variables	Definitions
T	Distance between front ping sensor and AP
L	Distance between back ping sensor and AP
M	Distance between front and back ping sensor
α	Angle formed between M and T
β	Angle formed between M and L
θ	Tether angle (AP orientation)
a	Front ping sensor signal amplitude
b	Back ping sensor signal amplitude
m	Front ping sensor gain constant
n	Back ping sensor gain constant
D	Tether distance
d_c	Conversation distance
d_1	Distance between wheelchair sagittal plane and receiver module.
d_2	Distance from AP sagittal plane to beacon.
i	Iteration number (program counter number)
j	Identification iteration number
K_{d1}	Front ping sensor low pass filter gain
K_{d2}	Back ping sensor low pass filter gain
θ_{low}	Low orientation threshold
θ_{high}	High orientation threshold
w	Window number
CMD_{UT}	Number of direction commands (*f, l, r, s*) (per window or total)
Dir_{AP}	Actual AP direction
t_{HIGH}	Tether distance high threshold
t_{LOW}	Tether distance low threshold
dir	Direction command produced by the UT system

3 Rule based algorithm to calculate wheelchair direction

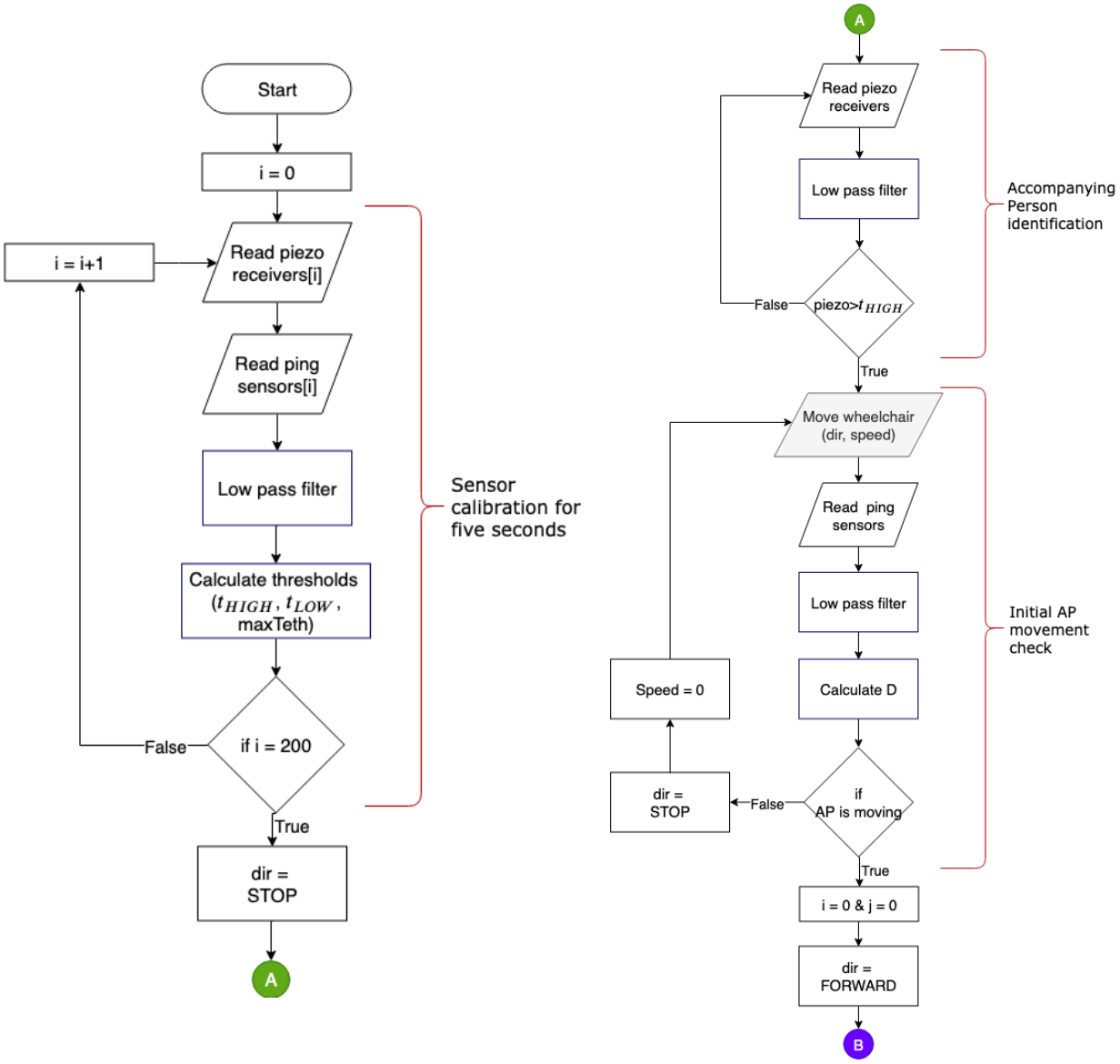


4 Beacon software flow diagrams

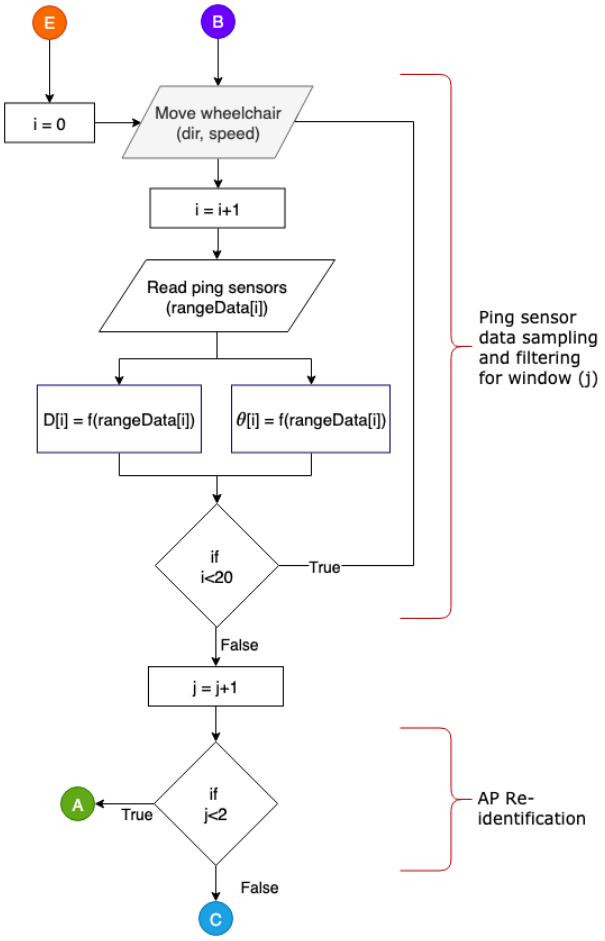




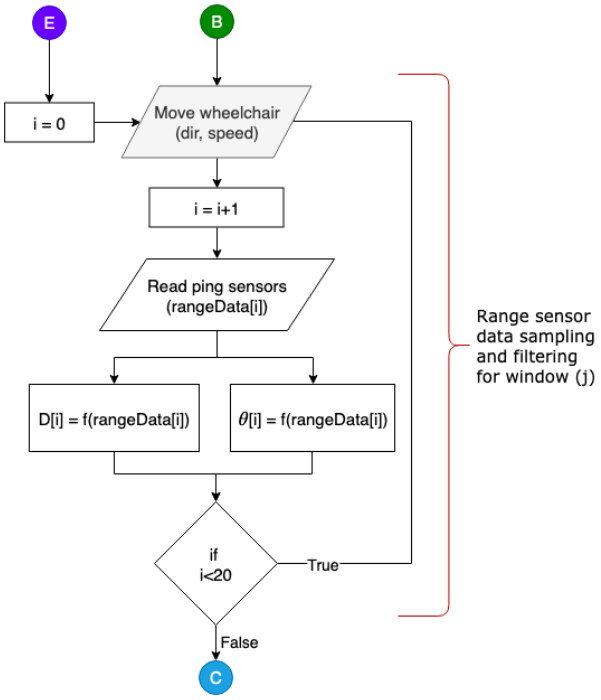
5 Receiver software flow diagram

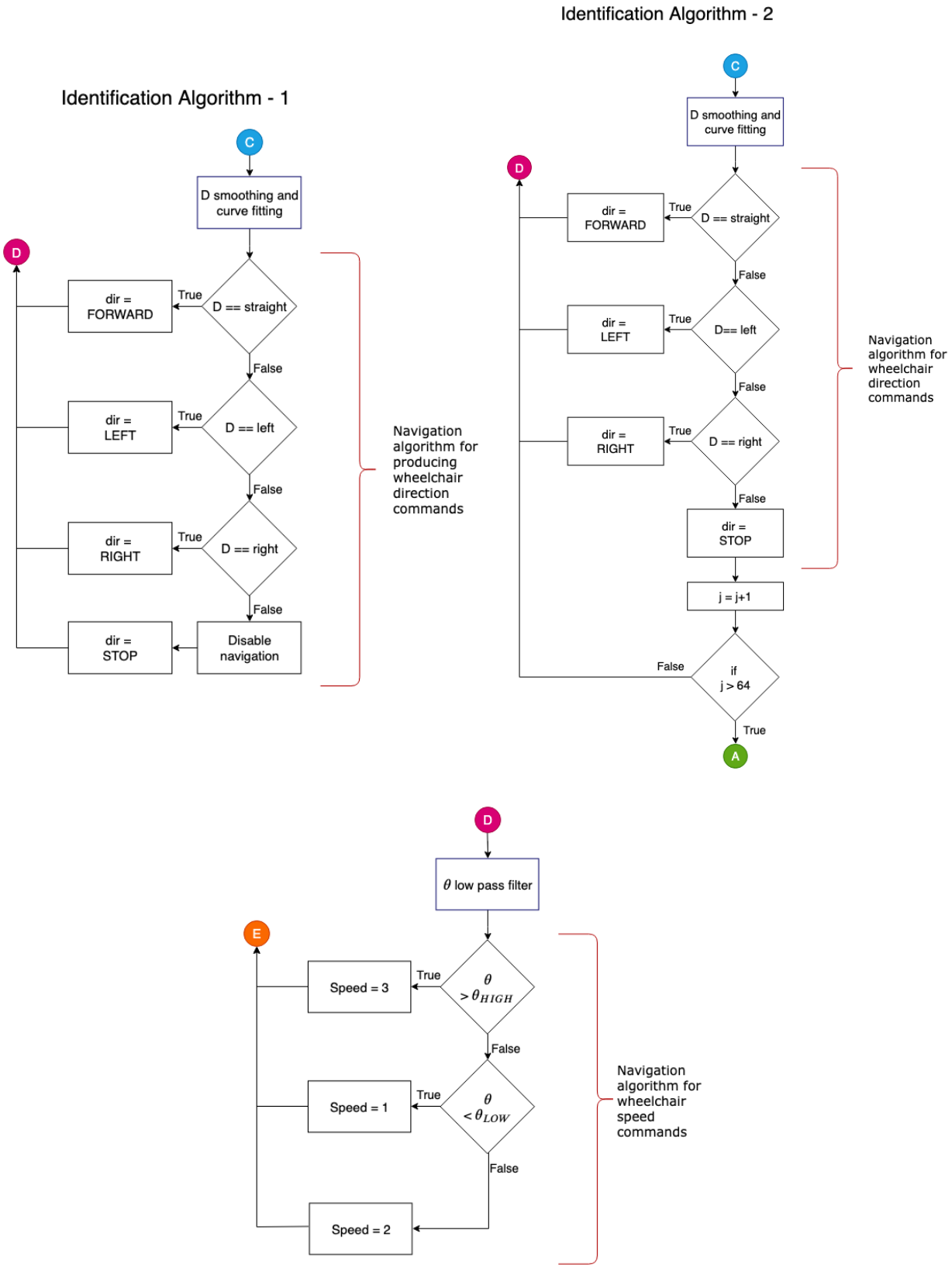


Identification Algorithm - 1



Identification Algorithm - 2




Identification Algorithm - 1
C
D smoothing and curve fitting
D
D == straight
True
dir = FORWARD
False
D == left
True
dir = LEFT
False
D == right
True
dir = RIGHT
False
Disable navigation
dir = STOP
Navigation algorithm for producing wheelchair direction commands
Identification Algorithm - 2
C
D smoothing and curve fitting
D
D == straight
True
dir = FORWARD
False
D== left
True
dir = LEFT
False
D == right
True
dir = RIGHT
False
dir = STOP
j = j+1
if j > 64
False
True
A
Navigation algorithm for wheelchair direction commands
D
θ low pass filter
E
θ > θHIGH
True
Speed = 3
False
θ < θLOW
True
Speed = 1
False
Speed = 2
Navigation algorithm for wheelchair speed commands