



Universidade do Minho

Escola de Engenharia

Maria Manuel Carvalho de Freitas Martins

**Online Control of a Mobility Assistance
Smart walker**

Dissertação de Mestrado
Ciclo de Estudos Integrados Conducentes ao
Grau de Mestre em Engenharia Biomédica

Trabalho realizado sob a orientação de
Professora Cristina P. Santos
Universidade do Minho
Professor Anselmo Frizera Neto
Universidade Federal Do Espírito Santo

DECLARAÇÃO

Nome: Maria Manuel Carvalho de Freitas Martins

Endereço electrónico: mariam@dei.uminho.pt Telefone: +351 965774086

Número do Bilhete de Identidade: 13460809

Título dissertação: Online Control of a Mobility Assistance Smart Walker

Ano de conclusão: 2011

Orientador: Professora Cristina P. Santos

Co-Orientador: Professor Anselmo Frizera Neto

Designação do Mestrado: Ciclo de Estudos Integrados Conducentes ao Grau de Mestre em Engenharia Biomédica

Área de Especialização: Electrónica Médica

Escola: de Engenharia

Departamento: de Electrónica Industrial

DE ACORDO COM A LEGISLAÇÃO EM VIGOR, NÃO É PERMITIDA A REPRODUÇÃO DE QUALQUER PARTE DESTA TESE/TRABALHO.

Guimarães, ____/____/_____

Assinatura: _____

Acknowledgments

First I want to thank my parents. Without their love, affection and all the support that they always gave to me in these past years, I may not be where I am now. In addition, for all their support, they always made available the necessary for my academic success, always instilling in me a great sense of responsibility. For these reasons I am very proud of them and hope never to have them disappointed.

Thank to my mentors, Cristina Santos e Anselmo Frizera.

Professor Cristina thank you for your unconditional support, knowloedge, experience and friendship. Thank you for all the strength and courage that you gave me so that everything went in the best way possible. Thanks to your constant encouragement, I could overcome all my problems and challenges.

Thank you Anselmo for the support and especially for allowing me to participate in this project and perform it in the best way. Even far we can be a good team and develop a great work.

To my colleagues from the CSIC group and ASBG for their friendship and help. I want to thank especially Eloy and Miguel that always shown willingness to help me in what I needed, always contributing with their good mood and knowlodge to help me to do a good work. Thank you also for the tremendous support and patience. And all the volunteers that performed the experiences.

To Ramón for your great knowldge and share of experience throughout this work.

To professor Paulo Carvalhal and Manuel João for the willigness, help and advices that you shared with me.

To my friends and brother who always accompanied me and lived with me this past year. Thanks for your support and patience and all the strenght that you gave me during this work.

To Sara Tribuzi for her enormous help in the formatting of this thesis.

Thank you very much to everyone.

Maria Manuel

Agradecimentos

As minhas primeiras palavras de agradecimento vão para os meus pais. Sem o amor, carinho e todo o apoio que sempre me deram ao longo dos anos possivelmente não estaria onde estou. Além, de todo o seu apoio, sempre me disponibilizaram o necessário para que o meu aproveitamento escolar dependesse apenas de mim, inculcando-me sempre um grande sentido de responsabilidade. Por estas razões tenho muito orgulho neles e espero nunca os ter decepcionado.

Aos meus grandes orientadores, Cristina Santos e Anselmo Frizera. Professora Cristina obrigada pelo seu apoio incondicional, conhecimento, experiência e grande amizade. Obrigada por toda a força que me transmitiu e ânimo para que tudo corresse da melhor forma. Graças aos seus constantes incentivos, mensagens de coragem, consegui ultrapassar as minhas dificuldades e superar os desafios.

Anselmo obrigado pelo apoio e força e principalmente por permitir que entrasse neste projecto, e que o realizasse da melhor forma. Mesmo longe conseguimos ser uma boa equipa e desenvolver um grande trabalho.

Aos meus colegas de grupo do CSIC e do ASBG pelo seu companheirismo e ajuda. Queria agradecer especialmente ao Eloy e ao Miguel que sempre mostraram disponibilidade para me ajudar no que fosse preciso, contribuindo sempre com a sua boa disposição para a obtenção de um bom trabalho. Obrigado também pelo grande apoio e paciência. E claro a todos os voluntários que me participaram nas experiências.

Ao Ramón Ceres pelo seu grande conhecimento e partilha de experiência ao longo deste trabalho.

Queria também agradecer ao professor Paulo Carvalho e professor Manuel João pela disponibilidade, ajuda e conselhos que partilharam para a realização de um bom trabalho.

Aos meus amigos e irmão que sempre me acompanharam e conviveram comigo neste último ano. Obrigado pelo apoio e paciência e por toda a força que me deram durante a realização deste trabalho. E claro à Sara Tribuzi por ter tido tanta paciência e me ter ajudado na formatação desta tese. Sem a tua ajuda isto não tinha ficado tão bonito.

“Pedras no caminho ?

Guardo-as todas, um dia vou construir um castelo.” F.P.

Abstract

This work presents the NeoASAS project that was developed at the Bioengineering Group, Consejo Superior de Investigaciones Cientificas (CSIC) in Madrid. Further, it continued with adaptations and improvements at Minho University with the Adaptive System Behavior Group (ASBG) in Guimarães, being designated by ASBGo Project.

These developments include the conceptual design, implementation and validation of Smart Walkers with a new interface approach integrated into these devices. This interface is based on a joystick and it is intended to extract the user's movement intentions. It was designed to be user-friendly and efficient, meeting usability aspects and focused on a commercial implementation, but not being demanding at the user cognitive level. Considering the ASBGo walker, the overall assemblage, mechanical adjustments, electronics and computing have been performed.

First, a review about the mobility assistive devices is presented, specially focused on Smart Walkers. Despite the intensive research, in current literature, there are not many works providing a "point of the situation", and explaining the role that robotics can play in this domain.

Healthy users performed preliminary sets of experiments with each walker, which showed the sensibility of the joystick to extract command intentions from the user. These signals presented a higher frequency component that was attenuated by a Benedict-Bordner $g-h$ filter, considering the NeoASAS walker and by a Butterworth circuit, considering the ASBGo walker. These methodologies offer a cancelation of the undesired components from joystick data, allowing the system to extract in real-time user's commands. Based on this identification, an approach to the control architecture based on a fuzzy logic algorithm was developed, in order to allow the control of the walkers' motors. In addition, a set of sensors were integrated on the walker for safety reasons: an infrared sensor to detect if the user is falling forwards; two force sensors to make sure that the user is properly grabbing the hand support; and two force sensors in the support forearms to verify if the user is with his forearms properly supported. This will make sure that the device stops when one of these situations happens.

Thus, an assistive device to provide safety and natural manoeuvrability was conceived and offers a certain degree of intelligence in assistance and decision-making. These results will be used to advance towards a commercial product with an affordable cost, but presenting high reliability and safety. The motivation is that this will contribute to improve rehabilitation purposes by promoting ambulatory daily exercises and thus extend users' independent living.

Resumo

Este trabalho apresenta o projecto NeoASAS desenvolvido no Grupo de Bioengenharia, do *Consejo Superior de Investigaciones Cientificas* (CSIC) em Madrid. Este teve continuidade com adaptações e melhorias na Universidade do Minho com o grupo *Adaptative System Behaviour* (ASBG) em Guimarães, sendo designado por projecto ASBGo.

Estes desenvolvimentos incluem o projecto concetual, implementação e validação de andarilhos inteligentes com uma nova interface integrada nestes dispositivos. Esta interface é baseada num *joystick* e tem como objetivo a extração de intenções de comando do utilizador, sendo intuitiva e eficiente. Atende a aspectos de usabilidade e está focada numa aplicação comercial, não sendo exigente a nível cognitivo. Considerando o andarilho ASBGo, foi realizada a construção deste, bem como, ajustes mecânicos, eletrónicos e programação.

É apresentada uma revisão sobre os dispositivos de assistência à marcha, tendo especial enfoque os andarilhos. Apesar da intensa investigação, na literatura não existem trabalhos que apresentem o ponto de situação desta área, bem como o seu papel na robótica de reabilitação.

Depois foram realizados testes com utilizadores, mostrando a sensibilidade que o *joystick* tem na identificação de intenções de comando do utilizador. Além disso, os sinais apresentam uma componente de alta frequência que foi atenuada, no caso do NeoASAS, com um filtro g-h *Benedict-Bordner*, e no caso do ASBGo, através de um filtro Butterworth implementado em hardware.

As metodologias apresentadas oferecem um cancelamento componentes indesejáveis, permitindo ao sistema a extração das intenções de comando do utilizador em tempo real. Desta forma, uma arquitetura de controlo baseada em *fuzzy logic* foi desenvolvida de maneira a fornecer uma assistência segura ao utilizador, através do controlo dos motores. Foram também integrados um conjunto de sensores no andarilho por razões de segurança: um sensor infravermelho para detetar a queda frontal do utilizador, dois sensores de força nos apoios de mão para detetar se o utilizador está a agarrá-los, e dois sensores de força nos suportes de antebraço para certificar que o utilizador está devidamente apoiado.

Assim, foi concebido um dispositivo que garante a segurança do utilizador e oferece um certo grau de inteligência e tomada de decisão. Estes resultados serão utilizados para a criação de um produto comercial com custo acessível, mas com alta confiabilidade. A motivação deste trabalho reflete-se na contribuição que este dispositivo terá na melhoria da reabilitação e desenvolvimento de dispositivos ambulatorios para promover exercicios diários, e melhorar a vida dos utilizadores.

Table of Contents

Acknowledgments	iii
Abstract	vii
Resumo	ix
Table of Contents	xi
List of Figures	xv
List of Tables	xxi
1. Introduction	1
1.1. Motivation	1
1.2. Context	1
1.3. Objectives	4
1.4. Results from the thesis	5
1.5. Structure of the thesis	5
2. Review and Classification of Assistive Mobility Devices	9
2.1. Introduction	9
2.2. Gait Dysfunctions	10
2.3. Mobility Assistive Devices	12
2.3.1. Mobility Training Devices	14
2.3.2. Self-ported Devices	19
2.3.3. External Devices	20
2.4. Walkers	21
2.4.1. Conventional Walkers	22
2.4.2. Smart Walkers	25
2.4.2.1. Physical Support	25
2.4.2.2. Sensorial assistance	28
2.4.2.3. Cognitive assistance	29
2.4.2.4. Health Monitoring	30

2.4.2.5.	Advanced Human-Machine Interface	30
2.5.	Conclusions.....	37
3.	Background and Presentation of the NeoASAS and ASBGo Projects	41
3.1.	Introduction	41
3.2.	Background: Project ASAS and SIOMBIOSIS.....	42
3.3.	NEOASAS walker and ASBGo walker	44
3.3.1.	Considerations about the NEOASAS and mechanical modifications	44
3.3.2.	Considerations about the ASBGo and mechanical modifications	46
3.3.3.	Subsystem for evaluation of user’s command intentions	48
3.3.3.1.	Specifications of the interface	49
3.3.3.2.	User’s gestures while guiding the Smart Walker	52
3.3.4.	Fall and Grasp Detection – Safety considerations	53
3.3.5.	Modular Architecture for Data Acquisition and Processing	53
3.3.5.1.	NeoASAS Architecture.....	53
3.3.5.2.	ASBGo Architecture	55
3.4.	Conclusions.....	56
4.	Validation of the Proposed Interface	59
4.1.	Introduction	59
4.2.	Evaluation of the signal related with the users’ command intentions in the NeoASAS and ASBGo walkers.....	60
4.3.	Joystick’s Signal Processing Architecture.....	67
4.4.	Signal processing architecture of the NeoASAS walker	67
4.4.1.	Selection and implementation of a filter strategy	68
4.4.1.1.	Selection of the filters parameters and their optimization.....	89
4.4.1.1.1.	Conclusion of the Filter Strategy.....	102
4.5.	Signal processing architecture of the ASBGo walker.....	102
4.5.1.	Y-signal Circuit Hardware.....	103

4.5.2.	Z-signal Circuit Hardware	105
4.5.3.	Digital Filter Strategy	108
4.5.4.	Digital Processing of the Joystick's signals.	109
4.6.	Conclusion of the Acquisition and Processing Architecture of the NeoASAS and ASBGo walker.	109
5.	Fall and Grasp Detection – Safety considerations	113
5.1.	Introduction	113
5.2.	Detection of a forward fall	113
5.2.1.	Walker's lower base	114
5.2.2.	Walker's upper base	115
5.2.2.1.	Sensor to detect the distance between the user's chest and the walker	115
5.2.2.2.	Detection of a forward fall	116
5.3.	Detection of a backwards fall.	120
5.3.1.	Detection of contact on the forearm supports	120
5.3.1.1.	Sensor to detect user's contact with the forearm supports.....	121
5.3.2.	Detection of handles grasping.	125
5.4.	Conclusions.....	125
6.	Control Strategy of the walker's motors	129
6.1.	Introduction	129
6.2.	Mechanical structure of the motor control unit.....	129
6.3.	Design and implementation of a control strategy to drive the walker.....	132
6.3.1.	Theory of the Fuzzy sets.....	132
6.3.1.1.	The Model.....	133
6.3.1.2.	Fuzzification	134
6.3.1.3.	Inference and Generation	136
6.3.1.4.	Defuzzyfication	137
6.3.2.	Design of the Fuzzy Inference System to control the walker's motors	138

6.3.3. Tuning, validation and results of the control strategy	144
6.4. Conclusion	153
7. Conclusions and Future Work.....	155
7.1. Conclusions.....	155
7.2. Future Work.....	157
References.....	159
Appendix A.....	167
A.1. Joystick and springs Datasheet	167
A.2. Y-signal Circuit Hardware	169
A.3. Z-signal circuit.....	171
Appendix B	173
B.1. Sharp Sensor Datasheet	173
B.2. Sensor Operation.....	176
Appendix C	178
C.1. <i>Flexiforce</i> Properties	178
C.2. Acquisition circuit of the flexiforce signal.	178
Appendix D	179
D.1 Motor's specifications (DOGA: 111.9041.30.00)	179
D.2. Control board used in the NeoASAS platform to control the motors.....	180
D.3. Control platform used in the ASBGo walker to control the motors	181
D.3.1 H-bridges (LMD18200, 3A, 55V)	181

List of Figures

Figure 2.1 a) Manual Wheelchairs [14] and Smart Wheelchairs with b) BCI interface [18] and c) bipedestation position [19].	13
Figure 2.2 Parallel Bars [20].	15
Figure 2.3 Robotic mobility-training devices: a) Lokomat [22], b) LokoHelp [26] and c) LOPES [27].	16
Figure 2.4 Ambulatory-training devices : a) LiteGait [34], b) KineAssist [32], c) Where-ll [33].	17
Figure 2.5 Standimovi mechanism.	17
Figure 2.6 a) GaitTrainer [35], b) HapticWalker [35].	18
Figure 2.7 NEUROBike [36]	19
Figure 2.8 Self-ported devices: Orthoses a)HAL-5 exoskeleton [40], b) ReWalkTM [41], c) RoboKnee [42], d) MIT active ankle-foot orthoses[44].	20
Figure 2.9 Passive Orthoses based in the Gravity-Balancing Principle [46].	20
Figure 2.10 External Devices: a) Standard Canes [59], b) Multi-feet cane [10], c) Crutches [50], d) SmartCane [48] and e) GuideCane [49]	21
Figure 2.11 Conventional Walkers: a) Standard walker [55], b) Front-wheeled walker [55] and c) Rollator [10].	22
Figure 2.12 Climbing the stairs with Boomer [53].	23
Figure 2.13 U-Step walking stabilizer [56].	24
Figure 2.14 Adjustment system of velocity and compactation of the U-Step Walker [56].	24
Figure 2.15 Smart Walker Móbil [64].	26
Figure 2.16 PAM-AID walker [70]	27
Figure 2.17 i-walker [72]	28
Figure 2.18 PAMM Smart Walker [74].	30
Figure 2.19 Walker with a joystick as an interface [77].	31
Figure 2.20 The haptic interface handlebars of the Morris's walker [58].	32
Figure 2.21 Guido Walker [78]	32
Figure 2.22 The Walkmate and its handle-bars as a force interface [79].	33
Figure 2.23 Prototype of the JARow [84].	34
Figure 3.1 ASAS walker [63].	42

Figure 3.2 SIMBIOSIS walker and a scheme of the orce sensors installed on the SIMBIOSIS walker [90].	43
Figure 3.3 NeoASAS walker.	46
Figure 3.4 a) Four-wheeled walker given by Orthos XX1 anf b) ASBGo walker.	47
Figure 3.5 Upper structure of the ASBGo walker.	47
Figure 3.6 Structure for the height-adjustment.	48
Figure 3.7 The joystick CH-Hall Effect.	49
Figure 3.8 Load Springs	50
Figure 3.9 Schematic of the joysticks' displacement.	51
Figure 3.10 The user is a) walking forward, b) turning left and c) turning right.	52
Figure 3.11 Schematic of the sensors integrated in a) the lower base and b) the upper base of the walker.	54
Figure 4.1 Typical raw Y (a) and Z (b) joystick data in the NeoASAS walker when the user is performing the three different trajectories explained in table 1: Top) A, Middle) B, Bottom) C trajectories.	62
Figure 4.2 Typical raw Y (a) and Z (b) joystick data in the ASBGo walker when the user is performing the three different trajectories explained in table 1: Top) A, Middle) B, Bottom) C trajectories.	63
Figure 4.3 Different phases of the trajectory B. S1- The user is stopped; S2- User starts walking forward; S3- User turns right; S4- User walks forward; S5 – User stops.	64
Figure 4.4 Typical raw Joystick data a) Y-signal and b) Z-signal obtained when the user is performing a translation curve (trajectory B) with the ASBGo walker.	65
Figure 4.5 Different phases of the trajectory C. S1- The user is stopped; S2- User starts walking forward; S3- User stops; S4- User walks forward; S5 – User stops.	66
Figure 4.6 Diagram of the processing architecture of the Y and Z-signals.	67
Figure 4.7 Example of a Frequency Spectrum for a) raw Y-signal and b) raw Z-signal. It is observed that user's command intentions are below 2Hz and have considerable more energy than the noise components (higher frequencies).	68
Figure 4.8 Attenuation of higher frequency components by means of recursive digital filters (Butterworth) with 2Hz of cut-off frequency. a) The raw Y-signal (black) and the filtered Y-signal (red) with the digital filter; b) and d) Frequency Spectrum of Y and Z filtered, respectively,; c) The raw Z-signal (black) and the filtered Y-signal (red) with the digital filter.	70

Figure 4.9 Influence of the variation of the g parameter on the filter BBF. a) $g=0.5$, b) $g=0.05$, c) $g=0.01$, d) $g=0.001$. All graphs represent the raw Z signal in black, the filtered Z signal with BBF in green and the ideal Z in red.	75
Figure 4.10 Influence of the variation of the θ parameter on the filter CDF. a) $\theta=0.5$, b) $\theta=0.95$, c) $\theta=0.99$, d) $\theta=0.998$. All graphs represent the raw Z signal in black, the filtered Z signal with CDF in green and the ideal Z in red.....	77
Figure 4.11 A complete Picture of the operation of the Kalman filter.	81
Figure 4.12 Measured state (raw Z) in black; estimated state (Z filt Kalman) in green and true state (ideal Z) in red.	83
Figure 4.13 Black: error between the true state and the measured state. Red: error between the true state and the estimated state.	84
Figure 4.14 a) The <i>a priori</i> error covariance (black) and the <i>a posteriori</i> error covariance (red); b) The kalman gain.	84
Figure 4.15 Measured state (raw Z) in black; estimated state (Z filt Kalman) in green and true state (ideal Z) in red.	85
Figure 4.16 Black: error between the true state and the measured state. Red: error between the true state and the estimated state.	86
Figure 4.17 a) The <i>a priori</i> error covariance (black) and the <i>a posteriori</i> error covariance (green); b) The kalman gain.	86
Figure 4.18 Measured state (raw Z) in black; estimated state (Z filt Kalman) in green and true state (ideal Z) in red.	87
Figure 4.19 Black: error between the true state and the measured state. Red: error between the true state and the estimated state.	88
Figure 4.20 a) The <i>a priori</i> error covariance (black) and the <i>a posteriori</i> error covariance (red); b) The kalman gain.	88
Figure 4.21 a) The superposition of the raw Z with the results of BBF, CDF and Butterworth; b) The superposition of the raw Z with the results of BBF, Kalman and Butterworth.....	100
Figure 4.22 a) The superposition of the raw Z with the results of BBF, CDF and Butterworth; b) The superposition of the raw Y with the results of BBF, Kalman and Butterworth.....	100
Figure 4.23 Attenuation of higher frequency components by the BBF filter a) The raw Z -signal, the ideal Z and the filtered Z -signal with the BBF filter; b) and d) Frequency Spectrum of Z and Y filtered	

with BBF, respectively,; c) The raw Y-signal, the ideal Y and the filtered Z-signal with the BBF filter. 101

Figure 4.24 a) Z-signal filtered with BBF filter ($g=16.87 \times 10^{-3}$) and b) Y-signal filtered with BBF filter ($g=44.29 \times 10^{-3}$) 101

Figure 4.25 Diagram of the circuit hardware for the processing of the Y-signal..... 103

Figure 4.26 a) The raw Y-signal processed by software (black) and the Y-signal processed by hardware; b) The raw Y-signal without processing. 104

Figure 4.27 Comparison between the Spectrum of the raw Y signal (blue)and the Y-signal filtered by hardware (red)..... 105

Figure 4.28 Diagram of the circuit hardware for the processing of the Z-signal..... 105

Figure 4.29 a) The raw Z-signal processed by software (black) and the Z-signal processed by hardware; b) The raw Z-signal without processing. 107

Figure 4.30 Comparison between the Spectrum of the raw Z-signal (blue)and the Y-signal filtered by hardware (red)..... 107

Figure 4.31 Attenuation of higher frequency components by means of recursive digital filters (Butterworth, first order). a) The Y-signal filtered by hardware (YH) in red and the YH filtered with the digital filter in black; b) Frequency Spectrum of Y and Z , respectively,; c) The Z-signal filtered by hardware (ZH) in red and the ZH filtered with the digital filter in black; 108

Figure 4.32 Complete system architecture implemented on NeoAsas walker. 110

Figure 4.33 Complete system architecture implemented on ASBGo walker..... 110

Figure 5.1 a) Sketch of the walker with the markers and b) the lower limbs of the user with the markers. 114

Figure 5.2 Frames from the BTS shootings. The lower base of the walker and the lower limbs in 2D. 114

Figure 5.3 Sharp IR sensor (GP2Y0A21YK0F) 116

Figure 5.4 Sharp IR sensor integrated in the upper base of the ASBGo walker. 116

Figure 5.5 Typical IR output signal in trajectory a) A, b) B and c) C..... 117

Figure 5.6 Typical IR output signal from user that is walking normally and then falls forward. ... 118

Figure 5.7 Algorithm to detect if the user is falling or not through the readings of the IR sensor output signals. 119

Figure 5.8 *FlexiForce* sensor..... 121

Figure 5.9 Mechanical device to support the sensor and ensure that the load made by the user on the forearm support is entirely made on the sensing area of the <i>Flexiforce</i> sensor.....	122
Figure 5.10 Assembly of the <i>FlexiForce</i> sensor with the forearm support.....	123
Figure 5.11 Typical signal of the <i>Flexiforce</i> sensor when first, the user is not loading the sensor, and then loads.....	123
Figure 5.12 Algorithm to detect if the user is falling or not through the readings of the flexiforce output signals.....	124
Figure 6.1 Motors on walker's structure of a) NeoASAS and b) ASBGo walkers.....	130
Figure 6.2 Processing platform, Board control and batteries located on the lower zone of the NeoASAS walker.....	131
Figure 6.3 Arduino Platform, H-bridges, sensors' auxiliary circuits and the power source.....	131
Figure 6.4 Membership function for a very negative value.....	133
Figure 6.5 Fuzzy logic system diagram.....	134
Figure 6.6 Examples of membership functions types [108].....	135
Figure 6.7 Inference and Generation steps of the Fuzzy logic system [108].....	137
Figure 6.8 Various defuzzification schemes for obtaining a numerical output [108].....	137
Figure 6.9 Membership functions for a) the forward input (Y), b) the rotation input (Z), and c) Motors. X-axis represents the numerical values of the inputs/outputs, and the Y-axis represents the degree of membership μ . These images are a print screen of the functions generated in Matlab.....	140
Figure 6.10 Scheme of the behaviour of the fuzzy system while the user is performing trajectory A.....	142
Figure 6.11 Scheme of the behaviour of the fuzzy system while the user is performing trajectory B.....	143
Figure 6.12 System Architecture and control flow of the NeoASAS walker designed in Simulink. The control block is presented in detail.....	145
Figure 6.13 Results from the system architecture of the NeoASAS walker acquired while the user was performing the trajectory A. a) raw acquired joystick signals and the result from the filtering with the BBF filter; b) signals before the amplification and restrictions; c) output of the fuzzy system (left) and the conversion and integration of these results to then be sent to the control board hardware (right).....	147

Figure 6.14 Results from the system architecture of the NeoASAS walker acquired while the user was performing the trajectory B. a) raw acquired joystick signals and the result from the filtering with the BBF filter; b) signals before the amplification and restrictions; c) output of the fuzzy system (left) and the conversion and integration of these results to then be sent to the control board hardware (right).148

Figure 6.15 A schematic of the complete control architecture system used in the ASBGo walker. It starts with a circuit hardware block that sends signals to the Arduino platform. Then the outputs are sent to the H-bridges, which power the motors.....149

Figure 6.16 Results from the ASBGo system of the acquired while the user was performing trajectory A. a) acquired joystick signals after the pre-processing with the hardware circuit. b) signals before the amplification and restrictions in the Arduino platform; c) output of the fuzzy system (left) and then the conversion and integration of these results to then be sent to the H-bridges and then to the motors (right).....151

Figure 6.17 Results from the ASBGo system architecture acquired while the user was performing trajectory B. a) acquired joystick signals after the pre-processing with the hardware circuit . b) signals before the amplification and restrictions in the Arduino platform; c) output of the fuzzy system (left) and then the conversion and integration of these results to then be sent to the H-bridges and then to the motors (right).....152

List of Tables

Table 2.1 Assistive Mobility Devices.....	13
Table 4.1 Description of the three trajectories (A, B and C) performed by five individuals, five times each.....	60
Table 4.2 SNR (signal-to-noise ratio) of the Y and Z signals (raw and filtered with the Butterworth filter).....	69
Table 4.3 Update and Predictions equations of the g-h filter.....	73
Table 4.4 Average values* of the best solutions of the g parameter (BBF), delay between the Y original signal and the filtered signal and KTE for each Y joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.	91
Table 4.5 Average values* of the best solutions of the g parameter (BBF), delay between the Z original signal and the filtered signal and KTE for each Z joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.	92
Table 4.6 Average values* of the best solutions of the θ parameter (CDF), delay between the Y original signal and the filtered signal and KTE for each Y joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.	93
Table 4.7 Average values* of the best solutions of the θ parameter (CDF), delay between the Z original signal and the filtered signal and KTE for each Z joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.	94
Table 4.8 Average Mean values* of the best solutions of the Q parameter (Kalman), delay between the Y original signal and the filtered signal and KTE for each Y joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.	95
Table 4.9 Average values* of the best solutions of the Q parameter (Kalman), delay between the Z original signal and the filtered signal and KTE for each Z joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.	96
Table 4.10 Best parameter solutions for each filter (BBF, CDF and Kalman).	97
Table 4.11 – Comparison between P initial value equal to the steady value, 0.5245, and P initial value equal to Q , using the average values of KTE and Delay, related to Y -signal, for each subject.	98

Table 4.12 Comparison between P initial value equal to the steady value, 0.5245, and P initial value equal to Q , using the average values of KTE and Delay, related to Y-signal, for each subject.	98
Table 4.13 Comparison between the proposed filters using the average values of the selected parameters for the 5 subjects for the Y signal. The presented average values correspond to the mean between the means of each trajectory .Table provides mean \pm standard deviation.	99
Table 4.14 Comparison between the proposed filters using the average values of the selected parameters for the 5 subjects for the Z signal. The presented average values correspond to the mean between the means of each trajectory .Table provides mean \pm standard deviation.	99
Table 4.15 SNR of the filtered Y and Z signals.....	102
Table 6.1 General Membership Funtions.	135
Table 6.2 Membership funtions of the inputs/outputs of the fuzzy inference system.	139

1. Introduction

1.1. Motivation

The number of people with reduced mobility capacities increases every year. The reduction on mobility is a factor that influences both people quality of life and their dependence of others in daily life.

This demands the development of devices that can support and aid this people. Among the existing mobility assisting devices, walkers play an important role on giving lateral stability, balance, partial weight support, among others, to its large number of potential users.

1.2. Context

In this context, it is presented a project of a robotic walker that was first developed in the Consejo Superior de Investigaciones Cientificas (CSIC) with the Bioengineering Group, in Madrid, called NeoASAS project. Further, it continued in Minho University with the Adaptive System Behavior Group (ASBG) at Portugal, being designated by ASBGo project.

These projects propose the development of a command and control interface of a robotic walker. It is based on an earlier project called SIMBIOSIS, in which was demonstrated the great rehabilitation and functional compensation potential of an instrumented walker. The SIMBIOSIS also allowed the conclusion that a new instrumentation system was necessary if a commercial focus was to be faced.

The purposed walker will give continuity to the SIMBIOSIS project, developed in CSIC, maintaining its stability and reliability, but with a simplification on the instrumentation system, giving the aim on a commercial focus. Therefore, this system has to be implemented with simple and low cost sensors that can provide for an active safety device and generate strategies for a more natural and secure command interface to guide the walker. This walker is based on a conventional four-wheeled walker.

This thesis focuses on the characterization of assisted human-machine interaction during gait, using a joystick and other safety low cost sensors systems. It is aimed at the development of an

interaction strategy between the user and the walker to control two separate motors that will push the device, in a safe and user-friendly way. The integration of this interface pretends to decrease the cognitive effort required to the user when he is guiding the walker.

The work developed in CSIC was coordinated by Dr. Anselmo Frizera and Professor Ramón Ceres and was divided into four stages that were implemented over the five months of work.

The first stage consisted on reviewing the existing literature about the assistive devices for human mobility, conventional and Smart Walkers, among others and human-machine interfaces. On the second stage it was performed an introduction to the programming oriented to the NeoAsas using RT Matlab xPC Target. Then it was made the assembly of the electronic to acquire and extract the signals from the joystick. On the third stage, the walker was experimented using healthy volunteers. The experiments with the users were done without any motorization of the device. This was done to acquire the natural signals obtained through the interface without the interference of any control strategy. So, by studying the signals that are capture from the interface the real movement intentions of the user can be seen. Then, with this information it was characterized the interaction between both agents. It was found that the signal was composed by two components: one related to the vibrations caused by irregularities on the device's wheels or the floor, and the other composed by transient events related to the guidance intentions. For each one of the components, a study to estimate them and/or eliminate them was necessary. Therefore, it was created a g-h filter to eliminate the vibrations from the signal.

After obtaining the data processing, a control strategy was implemented in the NeoASAS walker. This control strategy was based on fuzzy logic. The algorithm was developed to classify the signals sent by the joystick and transform them into motor inputs (direction and velocity). These algorithms were made in Simulink, MATLAB.

Finally, on the fourth stage it was made the validation of the control strategies of the guidance of the walker, adapting the programs to the electronic architecture. Necessary improvements were made during the validation. Then, new sensors and their localization were studied to improve the detection of dangerous situations.

This study was made with the help of a motion analysis system to analyse the position of the patient related to the walker, determining the possible places to integrate sensors that could detect possible falls of the user.

This study took continuity in Minho University, Industrial Electronics Department, under the supervision of Dr. Cristina Santos.

In the first stage of this work a new robotic walker was built, as well as its electronics. Through the mechanical modification of a conventional four-wheeled walker, the ASBGo was built. Thus, the first months were dedicated to the search of materials (steel, motors, pulleys, chains, sensors, forearm supports, handles, etc.) to the construction of the walker. A protocol was created with the company Orthos XXI that contributed with the conventional four-wheeled walker. Among the different types of walkers it was selected a four-wheeled to become possible the installation of sensors and motors for the driver of the walker. Among the four-wheeled walker it was given priority to one that allow the installation of the electronics and heavy components in the lower zone of the walker to improve the general stability of the set user-walker.

Then, it was searched for materials to modify the walker, as well as, DC motors, supports, handles, pulleys and others. When all material was found, the ASBGo was assembled. An additional structure was implemented to integrate the motors of the robotic walker and then an additional support base for the upper limbs, in order to find the best way to frame the interface that will be studied on the support.

The second stage consisted in surveying and extend the initial started state of art and write a journal review as well as an article about human-machine interfaces in Smart Walkers, including the work that is being developed in this thesis.

In the third stage, a study about the existing filter strategies suitable for this application, able to attenuate the noise components present in the joystick signals, was made, as well as experiments to find the best one to use here. The previously implemented g-h filter was more detailed studied as well as the Kalman filter. This stage involved both analytical, modelling and simulation phases.

The fourth stage consisted in designing the acquisition and processing architecture for the ASBGo walker. First, a hardware circuit to acquire, filter and amplify the joystick sensors was designed. Then, an Arduino platform was programmed with the control strategy studied in CSIC and adjustments were made. Finally, the connection to the motors through H-bridges was established.

The fifth stage began with experiments performed by 5 healthy volunteers. The joystick signals were acquired and analysed, as well as the signals sent to the motors. The validation of the designed hardware and the control strategy based in fuzzy logic was analysed and studied.

The sixth stage addressed additional safety issues. Firstly, research consisted in studying potential sensors to provide safety to the user during the use of the walker, to then select the more

suitable for this purpose. The decision about the sensors was based on price, precision measurement, type of power source and consumption, among others.

The idea was to find areas in the walker that could cover and identify dangerous situations, such as the backwards and forwards falls of a user. So these areas were studied and three were found: handles, forearm supports and upper base at the chest height.

In order to experiment and validate these areas, among some selected sensors, two types of sensors were selected: infrared sensors and Flexiforce sensors. These sensors were experimented, including experiments with users and its suitability was proven. The electronics and algorithm to classify the sensor signals were also developed and experimented.

At the end, a Smart Walker platform was built, with all the sensors interface integrated.

1.3. Objectives

In order to achieve the main aim of the proposed thesis, the following objectives have to be attained:

- Development and assembly of a Smart Walker platform based on a conventional four-wheeled walker and provide for the required modifications.
- Elaborate a critical review of the state of the art in the robotic technology for mobility assistive devices for people with mobility disabilities. The important role that robotics can play in mobility assistive devices is presented, as well as the identification and survey of mobility assistive devices subsystems with a particular focus on the walkers technology. It is presented a review of the available literature of walkers and it is discussed major advances that have been made and limitations to be overcome.
- The design of a robotic walker system. The sensory system must be integrated into an overall architecture to analyse the user interaction with the walker and to provide for his safety during the use of the walker.
- The development and implementation of control strategies, as well as the fusion of information of the sensors interface to detect the user's intentions so that he can guide the walker.
- The development and implementation of additional safety measures for the proposed

walker.

- To delineate and implement the required experiments to validate the proposal.

This work aims to build a walker with an autonomous system capable of controlling the movement of the walker through the extraction of the users' movement intentions obtained by the integrated sensory system. Thus, it seeks to develop a tool that provides for a clear benefit to a large number of people with limited ability to walk by improving the safety and reliability of walkers. Therefore, the users will avoid to inadequately resort to wheelchairs, that have disabling effects, thus contributing to the maintenance/improvement of the physical and cognitive capabilities of the user, through functional compensation.

1.4. Results from the thesis

A Smart Walker was built, implemented and validated, being a safe and stable device to support and help people with disabilities. In addition, a simple, user-friendly and intuitive interface was integrated into the device to allow an intuitive driven of the Smart Walker.

1.5. Structure of the thesis

This thesis is organized as follows.

In Chapter 2 it is presented a critical review of mobility assistance devices, with a special focus on the walkers as a tool for functional rehabilitation and compensation. This chapter has been proposed as an article review.

Chapter 3 does a brief presentation of the ASAS and SIMBIOSIS projects, followed by a presentation of the goals of the walker covered in this thesis.

In Chapter 4 it is described the main components of the signals obtained through the joystick and its potential to detect the intentions of the user to guide the walker. A processing strategy to improve/filter the signal is also presented.

In Chapter 5 it is described the additional safety measures included in this thesis and the characteristics of the applied infrared and *FlexiForce* sensors, as well as their potential to detect dangerous situations, such as, the fall of a user.

In Chapter 6 it is presented the control strategy implemented on the walker to attend the user movement intentions, as well as its validation on healthy subjects.

In Chapter 7 it is presented the conclusions and major contributions of this thesis, indicating possible areas of interest for future research on this area and to continue this work.

2. Review and Classification of Assistive Mobility Devices

2.1. Introduction

During the twentieth century the proportion of older people had an important rise, and this trend is expected to continue into the twenty-first century. The proportion of population over 60 years old was 8% in 1950, and 10% in 2000, and it is estimated to reach 21% in 2050 [1]. Losing complete or part of mobility, affects not only the ability to walk but also the ability to perform personal tasks, which is a major determinant in life quality and causes dependence of others in daily life.

In an aging society it is extremely important to develop devices, which can support and assist the elderly in their daily life, since their mobility degrades with age. This situation requires a great medical care, incurs large costs and can be fatal in some cases. Elderly tend to have cognitive impairments and experience more serious falls but there is strong evidence that daily exercise may result on fall prevention and postural stability [2]. So, it becomes more and more relevant to find ways and tools to compensate, to improve or to restore and to enhance this mobility.

For several years, researchers have been addressing the needs of persons with mobility disabilities alternative or augmentative devices [2]. These solutions are selected based on the degree of disability of the user. On the one hand, in case of total incapacity of mobility, alternative devices are used. These devices are usually wheelchairs or solutions based on autonomous especial vehicles [3-4]. On the other hand, the augmentative devices are developed to users with residual mobility capacities. They are used to avoid, whenever possible, the inadequate use of alternative devices, thus improving the physical and cognitive capabilities. These elements can be used as training mobility devices, self-ported devices, such as prostheses or orthoses, or external, such as crutches, canes and walkers.

Among the external augmentative devices, the walkers assume an important role, due to the large number of potential users, considering its simplicity and ambulatory potential. They were designed to improve pathological gait, through a support base for the upper limbs that improves the balance of the individuals and reduces the load on their lower limbs [5]. In addition, use the person's remaining locomotion capability in order to move [5], avoiding the early and deteriorative

use of wheelchairs. Over the past years, technological advances allowed the incorporation of sensors and actuators in conventional walkers, providing the stability of four-legged walkers, without affecting the resultant naturalness of the users' gait patterns. Besides, these devices enable to identify the movement intentions of the users and therefore control the mobility assistance accordingly.

It is intended to cover all of the major developments in the mobility assistive devices described before, particularly focusing on the walker devices. In this chapter, it is presented a brief review of the causes and consequences of the gait dysfunctions that lead patients to use mobility assistive devices. Then, it is reviewed the literature regarding the corresponding mobility assistive devices. Finally, the literature regarding the walker' devices is focused, presenting a more detailed review about the various existing models to date. It is presented a discussion of this information, summarizing the major accomplishments in the field and identifying the limitations to be overcome in future researches.

2.2. Gait Dysfunctions

The causes that affect the human mobility are rarely associated to only one disease [6]. The elderly are the ones that suffer more of mobility disorders, because the age causes a number of changes to mobility. Speed tends to decrease slightly and this is thought to increase efficiency of body motion since less energy is expended per stride. Less speed also allows better compensation of major muscle groups for any difficulties such as pain or weakness and allows less force to descend on any particular joint. There also tends to be some decrease in stride length. A decrease of speed and length may also play a role in maximizing balance and stability and be a natural way of helping prevent excess fall risk. [6]

The danger of falls in elderly people is due to high susceptibility to injuries caused by prevailing diseases e.g. osteoporosis or reduced protective reflexes. Besides the direct risk of fall-related injuries (e.g. major fractures or head trauma) another important consequence of falls is reduced mobility and loss of self-confidence, which in turn leads, therefore, to a significant reduction of quality of life. [7]

Disordered mobility, defined as a gait that is slowed, aesthetically abnormal, or both, is not necessarily an inevitable consequence of aging but rather a reflection of the increased prevalence and severity of age-associated diseases. These underlying diseases, neurologic and non-neurologic,

contribute to disordered mobility. Elderly patients usually have more than one condition contributing to their mobility disorder.

When asked about difficulties in walking, patients most often cite pain, stiffness, dizziness, numbness, weakness, and sensations of abnormal movement. Conditions seen in the primary care setting that can contribute to mobility disorders include degenerative joint disease, acquired musculoskeletal deformities, intermittent claudication, impairments following orthopedic surgery and stroke, and postural hypotension. Other conditions that cause mobility dysfunctions are hemiplegia, knee and hip diseases and metabolic disorders [6].

Walking is traditionally seen as an automatic motor task that requires little, if any, higher mental functions. In the past decade, new insights have drawn attention to the importance of cognition in daily walking. Normal walking requires strategic planning of the best route, as well as continuous interaction with the environment and with internal factors.

Failing to understand the significance of an obstacle, choosing an inappropriate route, or misinterpreting one's own physical abilities can lead to falls. The safety and efficacy of normal walking rely not only on sensorimotor systems, but also critically depend on the interaction between the executive control dimension (integration and decision of action) with the cognitive dimension (eg. navigation, visual-spatial perception, or attention) and the affective dimension (mood, cautiousness, and risk-taking). A common situation where such integration is challenged is observed when people must walk while performing one or more secondary tasks. Lundin-Olsson and colleagues [8] were the first to note the significance of a failure to maintain a conversation while walking ("stop walking while talking") as a marker for future falls. The ability to maintain normal walking while performing a secondary task has become the classic way to assess the interaction between cognition and gait.

In elderly people, this dual task ability deteriorates because central resources decline, secondary to subclinical disease processes or medication. This deterioration leads to a mismatch between the limited personal resources of elderly people and the complexity of the demand (the combined walking and secondary task). As a consequence, elderly people slow down or have increased stride variability (suggesting reduced automaticity) while performing a secondary task during walking. Gait becomes less secure and the risk of falling increases. In patients with overt disease, such as stroke or Parkinson's disease, gait deteriorates even more during dual tasking [9].

It is noteworthy, that the elderly are not the only age group to suffer from mobility disorders. Any person that has cardiovascular problems, strokes, etc., can be affected with the mentioned consequences.

To reduce the degree of mobility disorder, the patient can be exposed to a medical intervention. This intervention is effective, but usually leaves some residual damage that cannot be reverse. Thus, it is important to consider other types of interventions like the use of mobility assistive devices for rehabilitation or functional compensation of the mobility, so that the patient can recover from his dysfunction without injuries.

2.3. Mobility Assistive Devices

Because of the many available devices and the importance of selecting the one best for each individual patient, a formal evaluation should be carried out by an experienced physical therapist before purchasing a device. Once the device is purchased, the therapist can make the appropriate equipment adjustments to meet the specifications of each patient and can provide training in the proper use of the device. Potential problems can be assessed by the physical therapist before the device is used [10]. In Table 2.1 is presented the different categories of assistive mobility devices and the corresponding purposes.

In case of total incapacity of mobility, considering both bipedestation and locomotion, the alternative devices, represented by the wheelchairs and the special vehicles, are the optimal solution. These have been the targets of intensive research, especially considering Autonomous Robotic Wheelchairs (ARW) [11-13].

Beyond the manual wheelchairs (Figure 2.1a) [14], offer robotic solutions able to provide for autonomous and assistive navigation to wheelchairs (Smart Wheelchairs) making use of human-machine interfaces, like BCI's (Brain Computer Interfaces) and EMG (Electromyographic) signals [15-17] (Figure 2.1b). Additionally, some smart wheelchairs also enable a bipedestation position [19] (Figure 2.1 c)).

Table 2.1 Assistive Mobility Devices

	Mobility Assistive Devices	Examples	Purpose	Degree of incapacity
Alternative	Wheelchairs	<ul style="list-style-type: none"> • Manual • Autonomous with assistive navigation and/or bipedestation 	Transportation	Total incapacity of mobility
	Autonomous especial vehicles	<ul style="list-style-type: none"> • Autonomous vehicles to improve cognitive capacities; • Bipedestation 	Transportation. Improvement of cognitive capabilities.	
Augmentative	Mobility-training devices	<ul style="list-style-type: none"> • Parallel Bars • Treadmill-training devices • Ambulatory-training devices • Feet-manipulator training device 	Mobility Rehabilitation Training	Residual mobility capacities
	Self-ported devices	<ul style="list-style-type: none"> • Orthoses 	Functional Compensation (Supplement the function of the limbs)	
	External devices	<ul style="list-style-type: none"> • Canes • Crutches • Walkers 	Functional Compensation (Support during walking, to increase gait stability, and balance) and Rehabilitation training.	



a)



b)



c)

Figure 2.1 a) Manual Wheelchairs [14] and Smart Wheelchairs with b) BCI interface [18] and c) bipedestation position [19].

Nevertheless, the continuous use of this kind of devices can cause problems of health due to remain for a long time in a sitting position, such as loss of bone mass, osteoporosis, degradation of

blood circulation and physiological functions, skin sores, among others. For these reasons considering the remaining locomotion capacities, it is interesting to encourage, whenever possible, the use of augmentative devices [5].

The augmentative devices can help the patient with reduced mobility to avoid the previously presented health problems and allow using the user's patient remaining locomotion capability. In some cases, users can even relearn to walk safely and efficiently, in order to retrieve all the necessary movements for a normal gait. In other cases, these devices actuate like functional compensation elements that assist movements that the patient has lost. This means that the patient can learn new strategies to move, contributing to his independent locomotion on the daily life.

So, the augmentative devices can be used as 1) mobility-training devices during rehabilitation; 2) self-ported devices such as prostheses and orthoses, or 3) external devices, such as crutches, canes and walkers.

In the following, each of these types of augmentative devices will be described in more detail as well as the motivation for their use and evolution to more robotic systems when appropriate.

2.3.1. Mobility Training Devices

The mobility-training devices purpose is to help on the rehabilitation of the patient's movements that have gait disabilities.

The most basic mobility-training devices are the parallel bars [20] (Figure 2.2). Their objective is to help people regain their strength, balance, range of motion, independence and to recover from some injuries, and other debilitating conditions. Patients are required to repeat concise movements that work more than one joint and muscle. Parallel bars are also used for ambulatory exercises to improve a patient's ability to walk independently or with assistance. This kind of technique has shown to produce good results in terms of rehabilitation of patients [20].



Figure 2.2 Parallel Bars [20].

However, this therapy requires the involvement of two or three therapists to assist patients in walking, holding their lower limbs to control movement. Thus, it is required a substantial commitment and effort on the part of therapists [21]

Robotics appears applied to rehabilitation, resorting to the use of robots for mobility training of patients with disabilities.

It is known [22] that a new movement, or a qualitative improvement to an existing movement, can be achieved only through repetitions. Thus, therapy intensively addresses movement repetition in the training. This is the type of tasks that can be attempted by robotic devices. Research on this field led to the existence of robotic mobility training devices capable to expand the work of therapists so that each patient can receive enough therapy to easily attain an optimal recovery.

However, previous studies [23-25] have shown that patients have to be intensively engaged on the procedure; otherwise they lack interest with time, with a passive or no participation at all, causing bad results at the rehabilitation level. Their engagement must go beyond the novelty of the first sessions. Robotic mobility training devices are expected to provide to the therapists the necessary assistance to allow the patient to complete a certain task on locomotion; repeat as much as possible cyclic movements while motivating them to actively participate in the training.

Robotic mobility-training devices can be further divided into three different types of devices, according to the patients' pathologies.

The first are the treadmill-training devices, which apply a treadmill while training. Some examples use a partial body-weight support, assisting patients to re-learn which reduces the weight supported by the patient while training; therefore patients can re-learn walking movements through repetition and task-oriented training. The main aim is to eliminate the need of the patients to deal with balance during the gait, focusing their concentration on the lower limbs movements. The treadmill-based training devices have been the most common method for mobility training.

The target patients of these devices are neurological target users (stroke, spinal injury, cerebral palsy) [22]. Their training mainly aims at improving the functional movements and sensorial stimulation through repetition. On the literature, there are well known devices like Lokomat [22], LokoHELP [26], LOPES [27] (Figure 2.3), whose goal is to provide an intensive training on movement repetition.

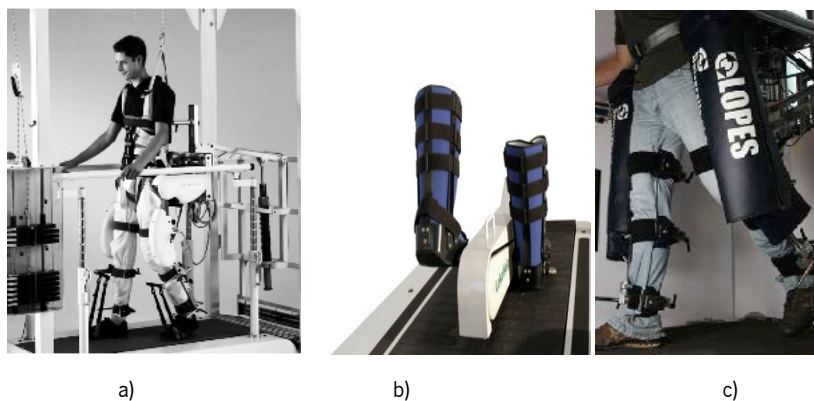


Figure 2.3 Robotic mobility-training devices: a) Lokomat [22], b) LokoHELP [26] and c) LOPES [27].

The second type of devices, denominated by ambulatory-training devices, is quite similar to the previous ones, but do not use the treadmill. Comparatively to the former type of devices, these propose an over ground training that can be as effective as training over a treadmill, with the advantage that requires less equipment.

Their target users are patients, which suffer from musculoskeletal or neurological disorders (muscular dystrophy, amputee, multiple sclerosis, spinal cord injury, lower extremity joint pain). One of the most debilitating aspects for musculoskeletal or neurological disorders is the loss of the ability to ambulate [28].

Ambulatory devices explore the ability of the device to provide dynamic assistance in order to facilitate the movement of the patient, so that the patient can learn to walk with proper upright posture. On the literature, there are examples like the LiteGait [29], Walkaround [30], WalkTrainer [31], KineAssist [32], Where-Il [33] (Figure 2.4) and Standimovi (Figure 2.5).

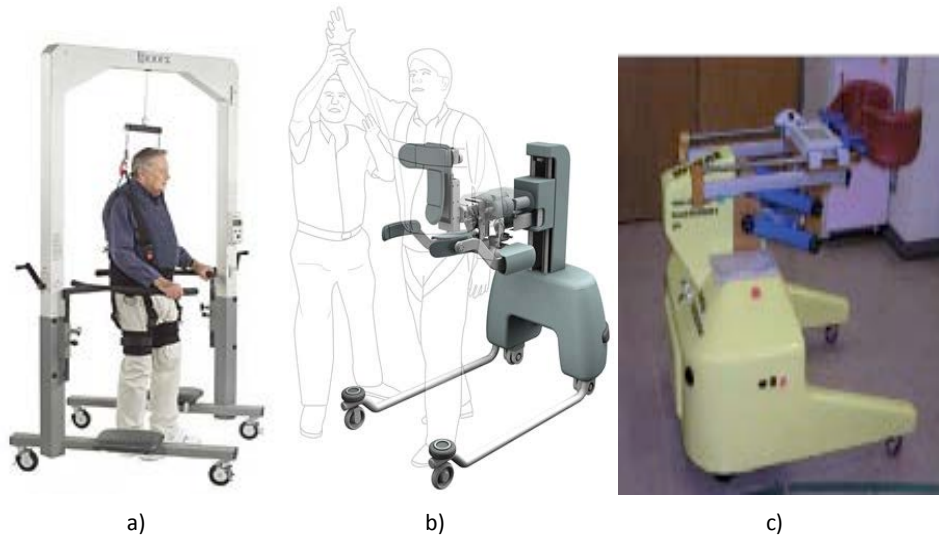


Figure 2.4 Ambulatory-training devices : a) LiteGait [34], b) KineAssist [32], c) Where-II [33]

Standimovi is currently in development at the Bioengineering Group – CSIC and the author of this thesis has actively participated in the first stages of the development and validation of its electronic platform.

This device is a self-guided ambulatory training device based on the integration of a human-machine interface to drive and adapt the device to the user’s requirements. Thus, this system aims to provide a higher degree of technical aid, to help patients to get autonomy to improve their remaining mobility capacity and mental state. Additionally, Standimovi also allows the patient to be transferred from the wheelchair to a walking position, without the help of others, with a safely and easily mechanism. The device and its mechanism are shown in figure 2.5.

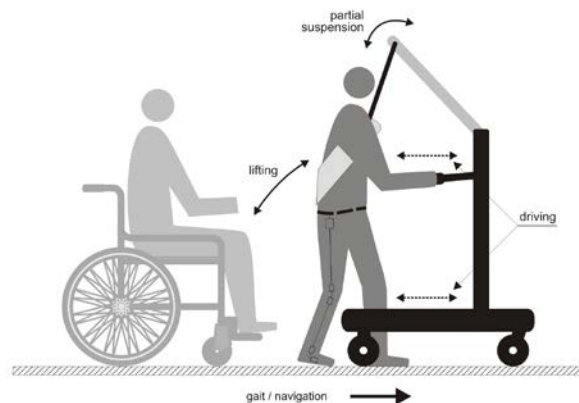


Figure 2.5 Standimovi mechanism.

Finally, the third type of devices denominated by feet-manipulator training devices, are based on holding the patient's feet to a robotic manipulator. It supports and gently leads the patients in the continuous practice of walking situations. The wheelchair-bound patient is attached to a steel frame in a type of harness, their feet on plates whose trajectories can be fully programmed and imitate everyday walking situations: walking on a level, tripping, slipping or climbing up and down stairs. Reacquiring these abilities is essential for mobility in everyday life. The objective is to copy these movements as naturally as possible. As a result of these artificial feet movements the slack muscles between the toes and the hips are forced into action again.

These devices address patients who have neurological disorders. In the literature it can be found examples like the GaitTrainer [35] and the HapticWalker [35] (Figure 2.6).



Figure 2.6 a) GaitTrainer [35], b) HapticWalker [35].

The described devices manipulate the legs, according to the kinematics provided by each mechanism and the speed desired. However, they need to be permanently installed in a room. They require that patients have to be moved from their beds to that room to experience the robotic rehabilitation. Unfortunately, depending on the healthy state of the patients, it may not always be possible to move the patient out of his bed to start the rehabilitation therapy. Additionally, waiting may imply destruction of cortical tissues, and less possibilities of recovering these neural functionalities [36].

To overcome the above-mentioned limitations, it was designed the NEUROBike [36] presented in figure 2.7. The NEUROBike system is a robotic device dedicated to the recovery of walking skills in stroke patients during the acute phase, when they are on bed, not yet able to keep a safe upright posture and walking. The system has been designed in order to provide, as soon as possible,

rehabilitation therapy, thus avoiding as much as possible further damages of the cortical tissues due to their non-use [37], and to facilitate neural function recovery with repetitive exercise.

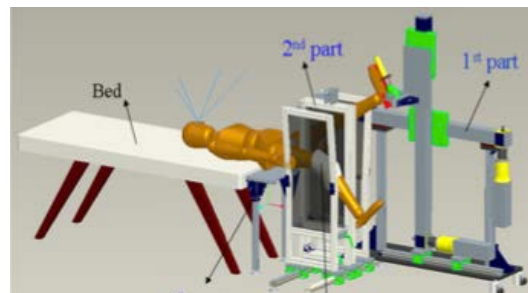


Figure 2.7 NEUROBike [36]

2.3.2. Self-ported Devices

Self-ported devices are carried by the user either to improve function of movable parts of the body (orthoses) or to substitute a lost member (prostheses) [38]. Between these two, orthoses are the ones that present an intense physical and cognitive interaction with the user, and are intended to offset the loss of mechanical function and work together with the movements of the patient. Orthoses allow the patient to perform a wide range of locomotor tasks on the ground, compared with the other mobility-training devices. Their function is to mechanically compensate or enhance functionally of the damaged member, acting in parallel with it [39].

Orthoses can be active or passive. Active orthoses bring the necessary energy to enable the movement by applying actuators or motors. These types of orthoses use actuators or motors. For a detailed survey verify work of AM Dollar [38] and N Costa [51]. HAL-5 (Hybrid Assistive Limb) [40], ReWalk™ [41], RoboKnee [42], MIT active AFO (ankle-foot orthoses) [43], MIT active ankle-foot orthoses [44] (Figure 2.8), and GAIT [45] are some examples that can be found in the literature.

Passive orthoses (Figure 2.9) don't provide energy, and remain passive. Energy is provided by the user. These devices only use springs and links and are based on the Gravity-Balancing principle [46]. This principle takes advantage from the gravity force to balance the gait of the patient.

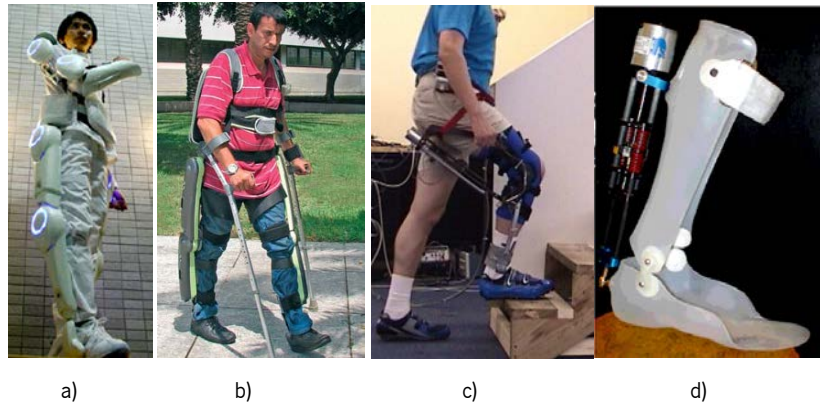


Figure 2.8 Self-porting devices: Orthoses a)HAL-5 exoskeleton [40], b) ReWalkTM [41], c) RoboKnee [42], d) MIT active ankle-foot orthoses[44].

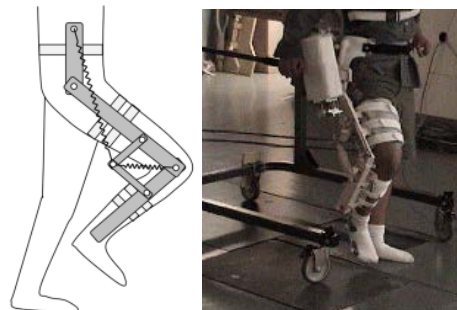


Figure 2.9 Passive Orthoses based in the Gravity-Balancing Principle [46].

2.3.3. External Devices

Canes, crutches and walkers represent the external devices. These devices are indicated to help users with balance problems and for partial weight-support.

Canes are more commonly used to increase the gait stability than to partial weight-support. A simple single point cane may prevent or reduce falls in patients with imbalance. An example is the standard cane [47] (Figure 2.10a). However, it does not seem to help falls due to retropulsion, which contributes to falls. So, there are canes that can provide greater support, like the multi-feet cane [47] (Figure 2.10b).

Crutches (Figure 2.10c) are orthopaedic devices that allow a direct support of the body thus providing great stability and balance in walking and a greater weight support compared with the canes. However, the crutches are cumbersome and are increasingly out of favour because they provide an unnatural gait.

As robotic examples of canes, it can be found the SmartCane [48] and the GuideCane [49]. (Figure 2.10d and figure 2.10e)

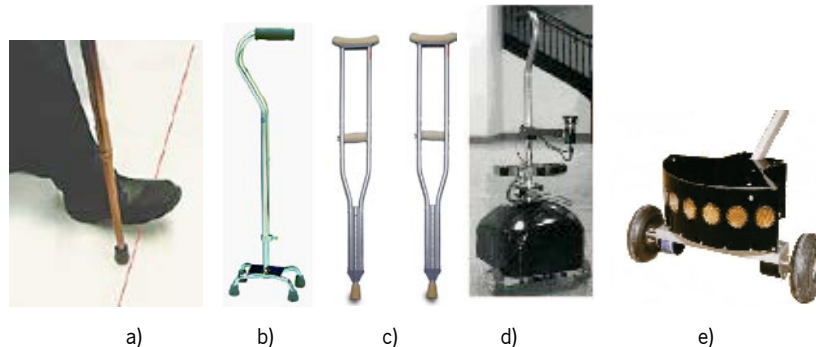


Figure 2.10 External Devices: a) Standard Canes [59], b) Multi-foot cane [10], c) Crutches [50], d) SmartCane [48] and e) GuideCane [49]

Finally, walkers are the devices from this group that provide greater support. They can have feet or wheels and are pushed by the user using hands or the forearms to guide the device. As this work is devoted to the study of these specific devices, it will be presented, on the following section, a detailed critical review about the walkers.

2.4. Walkers

Walkers assume an important role due to its simplicity and rehabilitation potential. These devices are interesting once they work as a supporting device during bipedestation and, in addition, use the person's own remaining locomotion capability in order to move [5] avoiding the early and deteriorative use of wheelchairs.

Walkers are prescribed to improve patients' mobility and help them maintain balance [10] [52]. The devices can increase confidence and sense of safety, which can raise a patient's level of activity and independence. There may be physiological benefits of limiting osteoporosis, reducing cardiopulmonary deconditioning and improving peripheral circulation [10].

Static equilibrium is maintained when the body's centre of mass is positioned over the base of support. Loss of balance can result when the centre of mass is displaced in relation to the base of support because of voluntary movements or external perturbations, such as slips, trips or pushes. Use of a walker increases the base of support, thereby allowing a greater tolerated range for centre

of mass [66]. They can also prevent instability by allowing stabilizing reaction forces such as holding on or pushing against the ground [10].

There are many types of walkers, considering their constitutive materials, accessories, sizes and structural configurations. These are classified them in two types: Conventional and Smart Walkers, detailed described on the following.

2.4.1. Conventional Walkers

An important aspect that classifies conventional walkers is the ground contact configuration, so they are classified by its type of support with the ground [5]: Standard, Front-wheeled and Rollators.



Figure 2.11 Conventional Walkers: a) Standard walker [55], b) Front-wheeled walker [55] and c) Rollator [10].

The Standard walker (Figure 2.11a) is a metal, four-legged frame with rubber tips, which must be lifted and moved forward while walking. This type of walker is used when maximum assistance with balance is required or when restrictions on weight bearing are present. While easier to use than a cane, this type of walker does require some degree of upper body strength and cognitive ability to use safely, and results in a fairly abnormal gait [61]. This device has also little value for retropulsion (many patients fall over backwards still holding their walker) and propulsion [10].

The Front-wheeled walker (Figure 2.11b) is designed for people who have weaker upper-limbs or that tend to fall backwards when lifting the device. This type of walker promotes a forward displacement of the center of gravity and allows a more normal gait, as the person can continue walking without stopping to lift the walker. It is particularly useful in patients with Parkinson's disease, as it reduces the risk of falling backwards. Cubo et al. [54] studied Parkinson patients who had gait freezing with objective measures and found that although standard walkers may stabilize

patients and may increase confidence, walking speed actually slowed when using a walker. In this study, the standard walker increased freezing and the wheeled walker had no effect on freezing.

In addition, it is less likely to allow the patient to pick up speed as he goes along, relatively to the four-wheeled walker [5].

The studies [10] show that most patients now prefer the Rollators (four-wheeled) compared to the other conventional walkers (Figure 2.11c). They do not need to be picked up, but have modern wheel systems so they roll and pivot smoothly and with little effort. They promote the most natural gait patterns. They are the easiest to use of the three types, but also provide the least stability. Wheeled walkers for use in the community can be equipped with baskets for shopping, and a seat that allows the person to stop and rest, if required [5]. Hand brakes can be selected to stop the device when squeezed or when released [10]. However, the user must be capable of learning to apply the brakes in order to use them safely [5]. Another option is the presence of a padded bar across the front that initiates braking if the patient presses or falls onto it. This bar is useful for patients with poor grip strength, speed or dexterity [10].

For example, Boomer [53], presented in figure 2.12, was designed by Daniel Molloy, Australia, and has a futuristic aesthetic. It allows the user to climb stairs: by pushing a bottom the front wheels can move along to near the back wheels so that it becomes a kind of cane that helps in the up and down of the stairs. The user can support his weight on the structure safely, because the wheels are locked and have a linear motion from top to bottom in order to help to climb the stairs.



Figure 2.12 Climbing the stairs with Boomer [53].

Another example is the U-Step walking stabilizer [56], of the In-Step Mobility (Figure 2.13). Its velocity can be adjusted through crank, as well as the height. Its structure can have also a compact form (Figure 2.14).



Figure 2.13 U-Step walking stabilizer [56].



Figure 2.14 Adjustment system of velocity and compactation of the U-Step Walker [56].

Despite potential benefits, looking at a variety of conditions in which assistive devices are prescribed, studies have shown that 30–50% of people abandon their devices soon after receiving them [67]. These findings raise questions about effectiveness, proper selection, appropriate training and potential problems.

Some reports have pointed out that walkers may actually increase the risk of falling by a variety of mechanisms [68][52]. A walker can potentially cause a patient to trip if it catches their foot. Similarly, a trip or fall might be precipitated by a device “catching” objects in the environment such as carpets, furniture or doorframes [69]. An unexpected perturbation of balance might occur if the device slips or tips over [69]. The potential instability of patients, like the ones with Parkinson, may make it more difficult for a patient to prevent a fall in these circumstances.

Recent research has pointed to the need for patients to allocate adequate cognitive resources to control an assistive device [52]. This may be challenging for patients with executive dysfunction, which impairs the ability to engage in more than one activity at a time and to switch attention and tasks, or dementia [10].

Other disadvantages of walkers include the need of more space in which to manoeuvre comparatively to a cane; difficulties on rolling on carpeting, and difficulties in crossing obstacles.

The use of any walking aid, in particular walkers, results in a slower gait speed and requires considerably more energy and cardiovascular fitness than walking unassisted. In this context, the assistance during the gait process is a clear application on the assistive robotic field [57].

2.4.2. Smart Walkers

Smart Walkers have emerged with the same structure as the conventional ones but they include additional robotic and electronic components, that promote a better assistance to gait, especially considering navigation [58], gait monitoring [59], and partial body weight support [59] [60]. However, some of these devices are too complex to use. Therefore, it is needed to take into account not only the users' disabilities in locomotion but also the fact that many of these users have additional deficiencies at cognitive and sensory levels. For instance, elderly people usually present slower behavior and are not familiar with mechatronic devices. Walkers should be designed to continually evaluate and correct their actions based on their perception of the needs of the user. Furthermore, the walkers need to be used during the daily routines of the user, such as, go to the bathroom, elevator, etc. Therefore, they have to be completely accepted by the user to their daily life.

Before building a prototype of a Smart Walker, it is needed to take into account usability issues, such as, safety, comfort and simplicity of use of the device. In addition, with the development and evolution of technologies it has been possible to integrate a range of useful features to the Smart Walkers, so these become easier to command and more comfortable to the user.

Thus, Smart Walkers have evolved to provide assistance to the user at different levels, depending on the user's needs. Generally, they can present the following functionalities [61]: i) Physical support; ii) Sensorial assistance; iii) Cognitive assistance; iv) Health monitoring and v) Advanced human-machine interface. This last functionality was proposed within this work.

In the following subsections these functionalities are presented along with a critical review regarding the most relevant Smart Walkers existing in the literature.

2.4.2.1. Physical Support

By its purpose, almost the totalities of the Smart Walkers present among their features some

kind of physical support functionality among their features to provide a better gait stability to the user. There are mainly two types of physical assistance: passive and active.

Passive physical support functionality introduces mechanical or structural enhancements to the device improving stability during gait. Usually, the improvements performed consist on the enlargement of the base of the device or the balanced placement of heavy elements (motors, batteries, electronics, etc.) at lower planes of the walker increasing the dynamic stability. Other passive change, not as commonly observed as these last ones is the replacement of the conventional handlebars of the walker by forearm support platforms [61][63]. This change has been introduced in the Mobil walker [64] (Figure 2.15), which was designed to offer more support assistance through the platform that supports the forearm, and additional help to the user when he is standing up or down.



Figure 2.15 Smart Walker Móbil [64]

The ASAS [63] and SIMBIOSIS [65] Smart Walkers have also used forearm support platforms. These supports were clinically tested with elderly and injured spinal cord users. These supports eliminate the degree of freedom of the elbow articulation and a higher fraction of the user's weight is supported by the device, making it easier to push and increasing the friction component of the system, reducing the risks of glide. Other possible enhancement concerning the reduction of gliding with the ground is the selection of materials with high coefficient of friction for the walker wheels.

Considering the walkers with three or four wheels, a common problem is the control of the device's free motion. More specifically, breaking a conventional walker is a task that requires muscular strength, motor coordination and good reaction time from the user, once these devices have usually a brake system similar to the installed on bicycles. If any of the before presented human faculties fail, there is a risk of an excessive acceleration of the device and a consequent fall.

In addition to the braking problems, it is important to emphasize that the strength and coordination are necessary to push/guide the walker can be high and thus difficult depending on the degree of disability of the user. In this case, it is important to provide external and controlled pushing energy to the system.

To prevent such situations active physical support functionalities are explored at most Smart Walkers [65] [70] [72]. Usually, these devices have motors installed on their wheels to control the brakes [70], to compensate gravity on inclined grounds [72] and to provide the pushing energy necessary to move the device [65] [71]. Different Smart Walkers explore different solutions to control wheels direction and velocity.

An example is the PAM-AID walker (Personal Adaptive Mobility AID) [70] [71], presented in figure 2.16, which aims to provide to the user the maximum control of the device during the entire time of its use in a more practical way. The controller of the PAM-AID just acts on the front wheels to control the direction of the device and it is composed by a handlebar that can rotate +/- 15° on the vertical axis. Besides, this walker has two operating modes: manual mode, where the control of the device is done only by the user, and assistive mode, where the system controls the front wheels to avoid obstacles.



Figure 2.16 PAM-AID walker [70]

Another example is the iWalker [72] (Figure 2.17), which incorporates inclinometers on its system that can detect if the surface is inclined or not. This information enables the walker to increase or decrease its velocity depending if it is an up or down slope.



Figure 2.17 i-walker [72]

Similarly to the aforementioned projects, the walker developed in this work will have support for the forearms, the heavy components will be at the lowest part of the walker, and the back-wheels will be motorized and controlled.

2.4.2.2. Sensorial assistance

The Smart Walkers may also provide sensorial assistance to the user, according to navigation and security issues.

Normally, the navigation and obstacles detection employ ultrasonic, vision or infrared sensors capable of detecting static and dynamic obstacles. The control system assists the user in obstacle avoidance by sound or vibration alerts or operating directly on the device's actuators, momentarily changing the path introduced by the user. This function is usually designed to help users with visual problems or to help navigation on environments with multiple obstacles.

The PAM-AID [70] (Figure 2.16) aims to guide blind people and is equipped with three types of sensors that are used to provide information of the surrounding environment: sonar, infrared proximity sensors and bumper switches. The information obtained by the environment is given to the user in the form of voice messages that describe the surrounding and warn about the emergence of obstacles. The data gathered is sufficient for obstacle avoidance and is simple enough to allow rapid processing. Another function is that the device can detect descending stairs, not dragging the user after it.

Similarly, the MARC Smart walker [73] includes a forward bumper and laser/infrared sensors to indicate the presence of an obstacle or stairs in the user's path.

Another example is the prototype that has been built on top of a Nomad XR4000 mobile robot platform [58]. This walker is equipped with two circular arrays of Polaroid ultrasonic transducers, two circular arrays of Nomadics infrared near-range sensors, three large touch-sensitive doors, and a laser range finder. These sensors enable our system to perceive obstacles at various heights and the laser range finder is used for navigation.

Early obstacle detection is extremely important for people who use walkers. Due to the effects of impaired balance, sudden changes in terrain can sometimes present serious challenges to balance, even when one is using an ambulatory aid.

However, none of these systems address the safety issue concerning about the risks of falls or other mishaps that could happen while the user is guiding the walker, or when the walker stops. They just evaluate the state of the environment and forget that the space area where the user is walking is also important to be aware of. So, it is necessary to develop subsystems that are alert to the situations of danger in that space. For this reason, the walker must have the capability to predict and avoid falls or similar situations that involve risks to the user and, then, stop. This is possible with the use of sensors that evaluate the gait of the patient relatively to the walker, like it will be done on this work.

Another problem is the possibility that the walker may roll away from the user while walking. This will be attended in this work by creating a sensing system, using infrared sensors, that measures how far the user is from the walker, detecting dangerous situations such as falls. Besides these solutions, it will be created a system with force sensors on the handlebars and in the forearms support that make sure the user is effectively guiding the walker with the two hands and supported by the supports, respectively.

2.4.2.3. Cognitive assistance

In this group, devices that assist user navigation and (auto-) localization in structured environments and outdoors (using GPS, for example) were considered. These Smart Walkers are very important to people that have cognitive issues and problems related to memory and orientation. Some Smart Walkers [72] [58] are programmed to follow predetermined paths inside clinical environment or to achieve a certain location in a map of a house or medical facility. Other devices are capable of creating maps of an unknown environment or auto-localization in a map using markers placed on the surroundings.

Smart Walkers may also be able to communicate bidirectionally with the user through a visual interface or voice commands, receiving directions from the user, or informing him about the present localization in a map and the environment conditions, e.g. obstacles.

The iWalker [72] (Figure 2.17), for example, has a navigation service that disposes to the user a map of the environment and their localization on it. They can ask for a route to reach some destination and real time indications to follow it. If navigation is interrupted by non-avoidable obstacles, the system can suggest a new route or offer to ask for help to a caregiver.

2.4.2.4. Health Monitoring

In more specific situations, the Smart Walkers can be used to monitor some health parameters of the user. This health information is used to keep a medical history of the user or inform through a wireless communication network a health centre or the medical staff in the case that an emergency situation is detected.

The PAMM Smart Walker [74] focuses specifically on the users' needs in an eldercare facility (Figure 2.18). It provides greater support for walking; monitors the health of the user (e.g. electrocardiogram), and it informs the user about his scheduled of tasks (e.g. taking medications). These functions were developed so that the elderly people can live more independently.



Figure 2.18 PAMM Smart Walker [74].

2.4.2.5. Advanced Human-Machine Interface

Systems or machines have increasingly evolved in capacity and functionalities, so now they can assist humans in tasks of higher level of abstraction. The elements that establish a bridge of interaction between human and machine, called interfaces, must establish a dialogue between these identities as user-friendly and efficient as possible [75].

The interface should be able to adapt to users with different levels of physical and cognitive capabilities and this adaptation should be done in a user-friendly manner.

In interfaces, the role of the sensors is very important since they measure the environment and human interaction and transmit the information to device so that it can accomplish the task that is being demanded by the user [76].

The Smart Walkers can present two different kinds of interface: direct interface in which the user commands or the user intentions can pass directly to the device; and indirect interface if otherwise.

One example of a direct interface is the joystick. Hashimoto et al. [77] developed a walker that is driven by a direct manipulation of a joystick (Figure 2.19). This joystick besides transmitting to the motors the user's intention of direction, it has a user-friendly hand force feedback system that allows the user to perceive obstacles and thus walk with greater safety. According to a repulsive force from the joystick, the user can know the distance and direction to the obstacles.



Figure 2.19 Walker with a joystick as an interface [77].

Force sensors have also been the focus of direct interfaces, because by detecting the user's intentions through physical interaction, the force sensor interface offers a user-friendly and intuitively way to reach that goal.

Handlebars are usually employed to provide for the support and stability in ambulatory devices. They require the user's hands to grip firmly. Morris et al. [58] developed a robotic walker based on the XR4000 platform with two-force sensors embedded into the handlebars. The idea is that by incorporating force sensors inside the handlebars, the user can maintain a steady hold and manipulate the robotic walker in a manner more consistent with contemporary roller-based walkers. The process runs as follows: each handlebar is equipped with a prismatic handgrip that is

motion constrained. Semi-pliable foam is inserted between the handgrip and motion stops to dampen the displacement exhibited by the grippers (Figure 2.20). A pair of force-sensing resistors is embedded into the foam to detect pressure when force is exerted along the handlebar. These pressure readings are transformed into planar translational and rotational velocities.

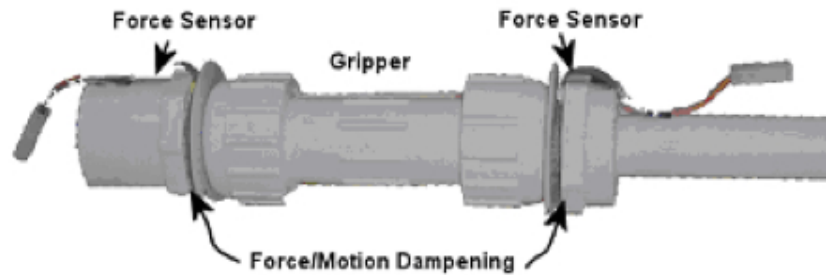


Figure 2.20 The haptic interface handlebars of the Morris's walker [58].

The latest version of PAM-AID, Guido [78], is guided by the user with spring-loaded handlebars, which are equipped with sensors to determine the intended direction of travel (Figure 2.21). Turn buttons are located on the end of each handlebar. Pressing these buttons causes the front wheels to turn parallel to each other in the same direction and thus allows the walker to rotate in a circle about its rear wheels. Brake levers are also positioned on the handlebars. If the user squeezes the brakes, the front wheels will both turn inward to stop the walker.



Figure 2.21 Guido Walker [78]

The Walkmate [79] has two force sensing resistors embedded into the handlebar (Figure 2.22). These sensors are fixed with the handlebar directly so it can detect both pull and push pressure of the handlebar. Moreover, the force sensors embedded into the handlebars offer a natural negative feedback loop of the motion control. When the user wants to keep a constant velocity, the handle

will be pulled to decelerate the walker if the walker is faster than the user. If the walker is slower than the user, the handlebar will be pushed to accelerate the walker.

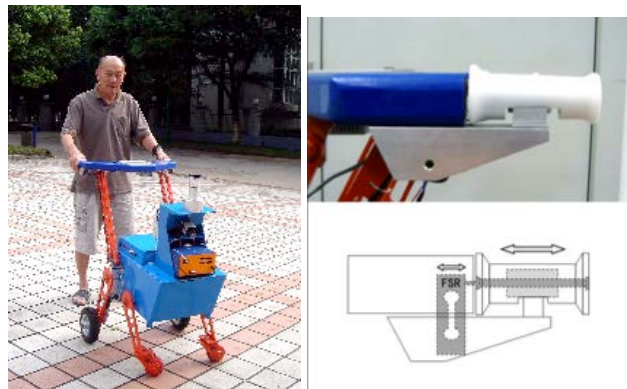


Figure 2.22 The Walkmate and its handle-bars as a force interface [79].

The ZJU Walker [80] is a motorized intelligent mobility aid that is being developed in Zhejiang University to help the elderly in the outdoor environment. It can extract a user's walking intent from the operations on the handlebars that include two torque sensors. The interface is natural and intuitive. The forces applied to the handlebars of the walker are transferred to the wheels' velocities through the dynamics of the walker, so with these sensors it can detect the directional, pace and rotary intent. For elderly users with degrading physical and cognitive functions, they may have difficulty in operating the handlebars appropriately; consequently the walker would not run as they thought. But the operations on the handlebars reflect their thoughts of how to operate the walker to some degree.

Other works addressing interfaces with force sensors are UTS [81], Chuy et al. [82], MARC [73] and SIMBIOSIS [65] and Sprint [88].

Another kind of direct interfaces is the voice communication. To help the visually impaired, for example, it was developed by Gharieb at King Saud University [83] a walker that has the ability to avoid obstacles and steer by following the user's voice. The walker receives the voice command from the user and determines the required tasks. If the speed of the walker exceeds a maximum limit value an audio alarm will be activated, and when there is an obstacle it warns the user with an audio alarm, too.

On the other hand, indirect interfaces have to recognize user's movement and/or intent without requiring manual operations. Examples include visual recognition using cameras [84], and detection of human gait using pressure sensors [85] and ultrasonic sensors [61].

A recent prototype of a walker is the JARow [84], which was developed to provide to the users the opportunity to recover from their disabilities in an efficient and economical way (Figure 2.23). The development of this walker took into account that elderly people have a very slow performance, and are not familiar with electronic and mechanical controls. Thus, the prototype JARow applies a pair of infrared sensors on the lower base of the walker to control the rotating speed of the motors. These sensors detect the position of the lower limbs of the user, which is used to calculate which direction and speed the walker must perform to follow the user.



Figure 2.23 Prototype of the JARow [84].

Research has mainly focus on developing direct interfaces with force/torque sensors, because unlike the joysticks, buttons and switches, for example, the force sensors are more robust, precise, and intuitive and provide better information. For instance, when the joystick is moved in any direction during the gait, due to the foot strike or uneven terrain, vibrations may appear.

Relatively to the buttons, switches and touch screens, they might be simpler and easy to control, but they have an intermittent or discrete operation that is mentally heavy for the user. Moreover, these interfaces may cause confusion for elderly users, which might result in an accident. The voice communication, for example, has an advantage of transferring effective high-level commands as a bilateral communication tool. However, there are critical problems such as interference and recognition yet to be resolved.

Indirect interfaces have strong demands on visual recognition, and to achieve progresses, it requires complex recognition algorithms with high computational costs and high performance devices.

Another problem is the human gait detection. Normally, these interfaces use ultrasonic or infrared sensors. In some cases, ultrasonic sensors require the user to worn an additional device

on the human body, which is a major disadvantage. Interfaces based on infrared sensors such as in the JaRoW prototype, besides being sophisticated, still have several issues to be resolved. For instance: the human gait control system is nonlinear; the gait parameters vary across users; the gait of users with disabilities is slow and irregular, and the sensors can mix the legs causing the controller to make wrong decisions about the intentions of the user.

Despite all these advances in the current state-of-the-art user-walker interaction field, there are still many unsolved questions and key areas in determining user-friendly and efficient interfaces. Further, it is very important to remember that these interfaces should not increase the cognitive burden or cause confusion and frustration to the elderly or to the lower limb disable users.

Recent studies on walker interfaces [77-87] have not focused on the characterization of the signals gathered by the interface sensors, and it is currently lacking an exhaustive analysis of the main parameters involved in the interaction between the user and the device. It is required to identify these parameters and their connection to the subsequent algorithms used for detection, attenuation, recognition and estimation. Different approaches may lead to a better manoeuvrability of the walker. Previous work [61] in which algorithms were developed to effectively estimate the cadence and significantly attenuate or eliminate the components of the force signals related to the oscillations of the trunk, achieving, thus, the intentions of the subject on the guidance of the walker. The approaches can be based on applying fuzzy control [61], model based-approaches [79] [88], probabilistic models [90] and others.

The Walkmate [79], for example, has two force sensors that detect both pull and pressure of the handlebar (Figure 2.22). These pressure signals are then sent to the computer to calculate the output to the motor control board. The velocity of the robot and the acceleration are set proportional to the force applied to the force sensor interface in the handlebars, which mean the robot will run or stops when the user's hands hold or leave the handlebars, respectively. However, this algorithm does not provide great stability to the guidance of the walker.

Another walker from MacLachlan et. al [81] that is being developed at the University of Technology, Sydney, takes two user intent variables from its force sensors: travel speed and turn rate. To measure user intent, two strain gauges have been placed on each handlebar of the walker based on FEM (Finite Element Method) analysis. Outputs from the individual circuits on each handlebar are combined to determine the user intent. The sum of the outputs gives the forward or backward moment on the handlebars and the difference gives the left or right turning force.

Another study from Chuy et. al [82] implements a control motion based on an applied force by the user that is passed to an apparent dynamics to determine the desired state of the system (velocity and position). Each desired wheel velocity is determined based on the inverse kinematic equation of the system. This implementation has shown to be instable to the system, so it requires a better evaluation of the control algorithm that uses the user's intentions to avoid instability. Model-based approaches require having a precise model of the walker, user and the surround environment and thus are prone to errors and have difficulty to adapt to variations, which are essential in real time applications.

Other examples of control systems can be found in [60] [80] [86-87].

The studies evaluated here lack a detailed description of the main parameters that describe the interaction between the user and the device, because it is necessary to improve detection and recognition algorithms used in interfaces.

It is therefore a key area to characterize the signals gathered by the interface sensors, by performing an exhaustive analysis and description of the main parameters involved in the interaction between the user and the device concerning the driving of the device according to the user's intention. In order to improve the reading of the users' intentions it is required to identify the parameters involved in these signals and their connection to the subsequent algorithms used for detection, elimination, attenuation, recognition and estimation. Further, due to the limitations on current recognition and control algorithms and computational costs that need to be decreased, there is room for further improvements and this is being a turning point for research.

In addition, the developed interfaces are uneconomic. It is noteworthy to call for the attention that the major prototypes [80-82] [84-85] were not evaluated with elder users or user with disabilities, which are in fact the target population.

Generally, the motion control algorithms are developed without taking into consideration the real interaction that happens between the user and the walker. To gather this knowledge it is necessary to run tests with the user using the walker without any control, and with the motors disconnected. With this it can be detected for sure the signals that represent the movement intentions of the user [90].

These experiments enable to collect the natural signals obtained through the interface without the interference of any control strategy, and thus to verify the real intentions of the user. This enables to develop a natural interface that does not require a learning phase or special skills to drive the walker.

So, the capture of signals that represent the intentions of the user will be captured by this explained form, so that it can be constructed the motion control algorithm to control the direction and velocity of the walker. This information will be applied to characterize the interaction between both agents.

2.5. Conclusions

On this chapter it was made a brief review of the dysfunctions that can affect the mobility, and was presented a critical review about the assistive mobility devices existing on the literature, giving emphases to the walkers, and more specifically the Smart Walkers.

The review about the walkers has shown that in the last years their development has witnessed a huge increase due to their enormous potential for rehabilitation and functional compensation. This potential emerges since this device makes use of the residual mobility capacities of users, maintaining and/or ameliorating them. Thus, it is provided the opposite effect of the wheelchairs, which have an incapacitative effect on the mobility of the users.

Evaluating the works that were presented in this chapter, one can infer that, in general, almost all the Smart Walkers have an assistive navigation system with sensors that detect obstacles, as well as, a design that has been studied to provide and improve a stable gait. This factor is possible with different designs of the guide platform that can have, for example, a forearm support. Another aspect that increases stability is the fact that electronics and other heavy components are put on the lower base of the walkers, giving a greater balance and stability to the device.

A general concern among Smart Walkers is the braking system that has to be easily manipulated, intuitive and effective, to avoid dangerous situations, such as, great accelerations of the walker on descending surfaces and the fall of the user.

The studies about interaction between the walker and the user are beginning to take place on the list of concerns with the Smart Walkers. This is an area that needs to be more explored because the interfaces that are being now studied, and that were presented on this chapter, are based on the direct supply of information from the sensors that detect the users' commands to the motors. There is a lack of description about what signals represent and what information they can give us about the commands of the users. This is necessary because the interface needs to be intuitive and natural to not require an excess of effort of the users' cognitive capacities. In addition, the interface has to adapt to all the users.

These developments have also a lack of detailed description of the main parameters that describe the interaction between the user and the device, requiring an improvement of recognition algorithms, as well as studies that evaluate the behaviour, gestures and posture of the user when is using the walker.

Future researches should improve the safety and stability of walkers, and avoid to inadequately resort to alternative devices such as wheelchairs, that have disabling effects, thus contributing and reinforcing to the maintenance or to the improvement of the physical and cognitive capabilities of the user. It is also necessary to better understand how the connection between the user and the walker works, as well as the benefits those walkers can bring to the gait, posture and motivation of the patient, to improve his results on the rehabilitation process.

3. Background and Presentation of the NeoASAS and ASBGo Projects

3.1. Introduction

The critical review about mobility assistive devices, held in Chapter 2, highlights a considerable number of mobility assistive devices, including Smart Walkers that are being developed throughout the world. This highlights the importance of this matter today.

The research on walkers has been focused on implementing assistive navigation and driven control, not focusing on the use of forearm support, manoeuvrability, safety and the cost of the walker. Further, the focus is generally on the evaluation of the walker surrounding, not paying due attention to the area where the user is walking.

These considerations have, however, been addressed recently. The Smart Walkers proposed in the NEOASAS and ASBGo projects, aim to provide for safety, a natural manoeuvrability and try to offer a certain degree of intelligence in assistance and judgement. Information gathered by several sensors is used to inform the Smart Walker about the status of the user, i.e. identify the movement intentions of the user considering direction and velocity, as well as additional safety, by detecting possible falls of the user.

Results will be used to advance towards a commercial product with an affordable cost but high reliability and safety. The motivation is that this will contribute towards better rehabilitation purposes by promoting ambulatory daily exercises and thus extend users' independent living and improve their life quality.

The NeoASAS project began on CSIC and its continuation is taking place in University of Minho with the ASBGo walker, under the work proposed in this thesis. NeoASAS emerges as the evolution of two projects, ASAS (Andador Pseudorobótico de Alta Seguridad) [63] and SIMBIOSIS [61], previously developed in CSIC. In this chapter, it will be done a brief presentation of the ASAS and SIMBIOSIS projects, followed by a presentation of the goals of the NeoASAS and ASBGo projects covered in this thesis.

3.2. Background: Project ASAS and SIOMBIOSIS

The Project ASAS [63] of the Bioengineering Group at IAI – CSIC aims to develop a Smart Walker to the rehabilitation and the maintenance of the natural mobility [63].

ASAS (Figure 3.1) is a modification of a commercial device in which the base has been enlarged, the handles have been replaced by a forearm-supporting base and the heavier components have been placed on the lower plane close to the ground. These modifications were made to increase its stability during the gait of the user. It has been also installed self-blocked dc motors on the rear wheels, pushing energy to the displacement of the walker and braking control. The control of the motors is made through push buttons installed on each handle.



Figure 3.1 ASAS walker [63].

Although the good results achieved with this walker during clinical evaluations [91], the system presented poor user-interface. The fact that each motor is controlled by one push-button makes the control of the walker unnatural to the user. The project evolved onto the SIMBIOSIS project.

SIMBIOSIS is a Smart Walker [61], presented in figure 3.2, and is intended to be guided by the user's intentions, according to his gestures and postures. In order to do this a series of sensors acquire and process the required interaction parameters.

The user interface of the SIOMBIOSIS walker is divided in three subsystems: i) feet/user relative position, ii) lower-limb kinematic and iii) upper-limb force interaction.

Subsystem (i) is compounded by ultrasonic sensors and it was designed to obtain the temporal evolution between the feet of the user and the walker. [92] The information acquired enabled to analyse the gait during the use of the device, providing information related to the walker's velocity

and also related to safety such that the device is stopped in case of excessive separation or approximation to the walker, diminishing the risks of falling.

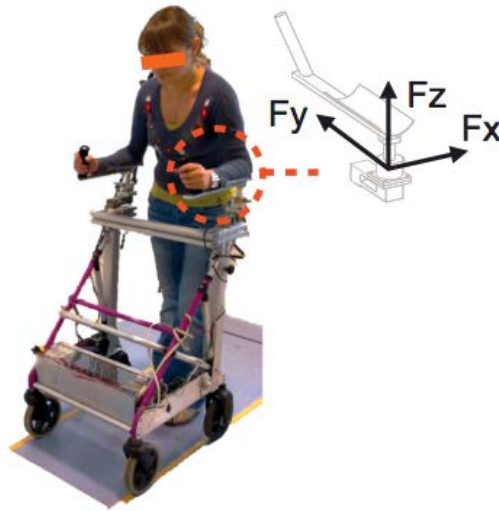


Figure 3.2 SIMBIOSIS walker and a scheme of the orce sensors installed on the SIMBIOSIS walker [90].

Subsystem (ii) is compounded by a series of inertial sensors placed along each leg of the user. Each inertial sensor enables to estimate the orientation and evolution of the lower limbs on the space.

These sensors constitute a portable gait analysis system to determine parameters related to the gait phases, and also spatial-temporal parameters, velocity and step lengths, as well as the irregularities in the gait process. All this information is relevant to inform if the gait parameters of the user improve with the use of certain control strategies.

Finally, subsystem (iii) is compounded by a series of force sensors installed on the forearm-supporting base (Figure 3.2). This subsystem is used to study the interface of the forces of interaction between the user and the walker during the gait, in order to extract patterns during the execution of certain tasks with the device. These patterns are then used to infer the intentions of the user's navigation, defining the basic commands for the robotic system [90].

The results obtained with this project were quite satisfactory. It was possible to increase and characterize the load due to user's partial body weight support to improve the gait patterns and to increase the stability and safety of the gait. Testes performed with paraplegic at the Spinal Cord Injury Hospital of Toledo (Spain) demonstrated the large rehabilitation and functional compensation potential of the developed walker. [93] These two projects, ASAS and SIMBIOSIS, allowed to verify that a new sensorial system was necessary if a commercial focus was envisage. The NeoASAS

Project emerged in order to give continuity to the ASAS and SIMBIOSIS projects, but with a commercial focus. It is intended that the NeoASAS walker has the same reliability and stability that the SIMBIOSIS walker [61], using a sensor system for the characterization of the user-walker interaction during assistive gait. Nevertheless, in order to make it more robust and considering economic aspects it requires a simplification of its sensorial configuration. Thus, the device should be implemented with simple and low cost sensors. However, the interface of this new prototype must remain user-friendly without requiring additional cognitive capacities or training, since many users have many difficulties related to these factors. These were the problems tackled in this thesis that fits in the NEOASAS project. In Minho University these goals were also implemented in the ASBGo walker, as well as other considerations related to the security of the user while he is walking with the device.

The NEOASAS and ASBGo walker, in their final state, will be an autonomous system capable of controlling the movement of the walker through the extraction of the users' movement intentions obtained by the integrated sensory system. These projects are unique in the sense that they try to improve the safety and stability of walkers, and avoid to inadequately resort to alternative devices such as wheelchairs, that have disabling effects, thus contributing and reinforcing to the maintenance or to the improvement of the physical and cognitive capabilities of the user, through functional compensation.

The NEOASAS and ASBGo goals directly related or addressed in this thesis will be detailed in the following. These comprise mechanical modifications to a standard four-wheeled walker, additional sensorial subsystems and cognitive improvements onto the control architecture.

3.3. NEOASAS walker and ASBGo walker

3.3.1. Considerations about the NEOASAS and mechanical modifications

The use of four-wheeled walkers is increasing in a number of countries around the world and their users seem to be satisfied with their assistive device. Because of this information and according to the following considerations, the NeoASAS walker was chosen to be a modification of a four-wheeled walker [61]:

- It enables the user to achieve a more natural gait, since it is not necessary to lift the device to walk.
- It enables a broad base of support, since there is always contact between the walker and the ground.
- As it was presented in [94], it was observed a reduction of reaction time to external stimulus when comparing the use of a four-wheeled walker with a standard walker.
- In the study presented in [95], it was observed a 50% reduction of the energy consumption when using the four-wheeled walker compared with the conventional walker with four legs.

Besides these considerations, were analysed the disadvantages/limitations of the four-wheeled walker [52]. This walker presents problems concerned with handling and manoeuvring. For instance, users with reduced strength on the arms have many difficulties on driving the four-wheeled walker [52]. Other problems are relative with the brake system that usually requires strength the users do not have and the brake cables are loose. Moreover, the four-wheeled castor wheels revolve and can get stuck; the suspension is poor as well as the directional control, which transmit insecurity to the user, as well as instability to when he is guiding the walker.

In order to tackle these problems, the NEOASAS walker (Figure 3.3), was integrated with two support platforms, for the hands and forearms, which allow a greater unload of the partial body weight of the user on the walkers' structure. This reduces the load on the lower limbs and extends the use of the device to individuals that have muscle weakness, joint pain and certain types of lesions. At the same time, it is possible to reduce the uncontrolled movements of the wrist since this is weak, difficult to control and prone to errors. Additionally, it was integrated in each rear wheel a DC motor that allows the control of the walker movement and braking, resulting in a more secure and stable walker.



Figure 3.3 NeoASAS walker.

Beyond these mechanical modifications it has been installed on the walker a user interface based on a joystick, that measures the commands of the user (navigation intentions related to direction and velocity), so that it is easier for the user to guide the walker. The joystick is intended to substitute the force sensors of the SIMBIOSIS, since it is a simpler sensor with a lower cost.

3.3.2. Considerations about the ASBGo and mechanical modifications

At Minho University the concepts presented in the NeoASAS walker were reproduced and a four-wheeled walker (Figure 3.4a), given by Orthos XX1, was modified to build the ASBGo walker (Figure 3.4b)). This walker has the same mechanical modifications as the NeoASAS walker, as well as the joystick interface, but with some improvements:

- The upper base of the NeoASAS walker, where the joystick is integrated, is made from steel that bends when the user pushes it to guide the walker. This introduces errors on the reading of the movement intentions made by the joystick. In the ASBGo walker a stronger material, iron, that does not bend when the user forces it, was applied (Figure 3.5).
- The joystick was not well fixed on the upper base of the NeoASAS, so some intentions were not well read. In the ASBGo it was introduced a plate that forces the joystick to

follow the movements of the upper structure, as well as a rubber that fixes the joystick onto the plate (Figure 3.5).

- The upper structure was not well aligned in the NeoASAS walker, being other reason to introduce errors and noise to the joystick. While the ASBGo structure was built a special attention to these alignments was paid.
- The height-adjustment system was difficult to manipulate in the NeoASAS, and now the ASBGo has a better one and easier to adjust (Figure 3.6).
- The supports of the forearms are more comfortable in the ASBGo walker (Figure 3.4b).



Figure 3.4 a) Four-wheeled walker given by Orthos XX1 and b) ASBGo walker.

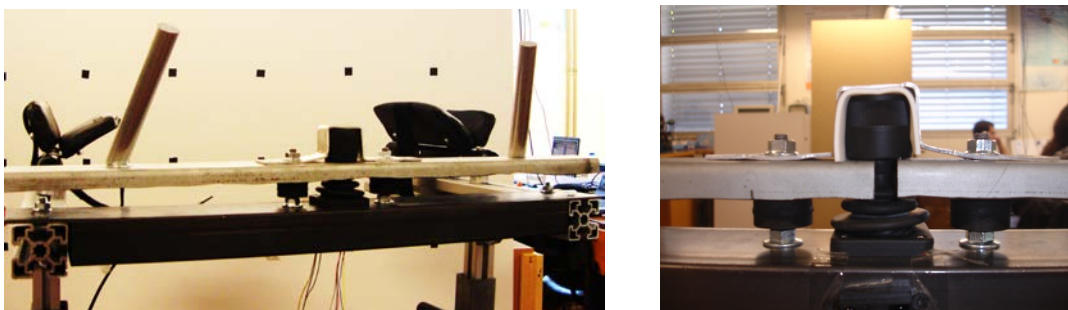


Figure 3.5 Upper structure of the ASBGo walker.

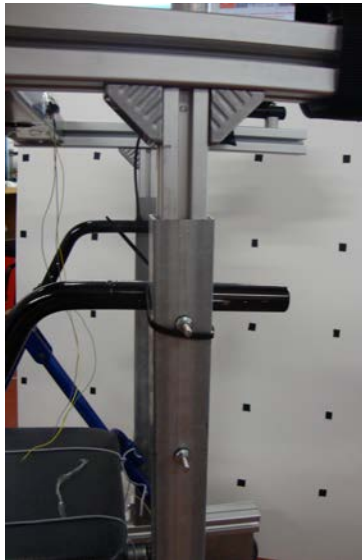


Figure 3.6 Structure for the height-adjustment.

Additionally, an infrared sensor has been added to the ASBGo walker at the user's chest height to detect if the user is falling. Two force sensors have been placed on each forearm support to verify if the user is correctly supported and other two force sensors have been added to verify if the user is grabbing properly the hand support. These sensors have been introduced for safety reasons. The author of this thesis suggested all these considerations.

3.3.3. Subsystem for evaluation of user's command intentions

In order to provide for walking support, the NeoASAS and the ASBGo have to capture the direction that the user intends to go based on the user manipulation. The directional intent will be identified from physical manipulation because the user's directional intent and his physical manipulation usually are mutually consistent. In this work, a novel interface is proposed to recognize the user's directional intention according to the manipulation of handles on the upper base support.

The development of this interface took into account that walking is a complex combination of omni-directional motions, including forward, translational (right and left), oblique motions and rotations. The interface must have the capability to "read" and interpret all these kind of intended motions, to follow the user's movement, and to provide for a good walking support. Additionally, the novel interface has the advantage of using the forearm supports, improving safety during

walking and helping to unload the lower limbs. Moreover, the user does not have to intentionally manipulate the device in order to set direction instructions and it is low-cost, if compared to other commonly used technologies in this field, such as force sensors or Brain Computer Interfaces (BCI's).

The system proposed for reading the intentions of the user's walking movement is based on a joystick that is integrated into the upper base support at its middle. This joystick is the key of the interface.

3.3.3.1. Specifications of the interface

During the SIMBIOSIS project it were observed and studied which efforts were the most important to drive the walker. According to these studies, in the NEOASAS and ASBGo project were proposed to use a new interface that consists on placing, at the centre of the upper base support, a joystick associated to a spring that is moved according to user's manipulation (Figure 3.5). When the user forces the handles, this slight movement is transmitted to the upper base support, mechanically coupled to the joystick that reads the user intention. When the user begins his gait, he has to slightly move the handlebar through the handles, moving the joystick, informing the walker which direction and velocity he wants to take. Hence, the user's efforts are successfully converted into small movements through this new interface, which is simpler, but is developed to be optimized for commercial implementation.

The handlebar is a base and in its centre is integrated the joystick associated to springs, and in each one of the two extremes was placed a handle, so that the user can hold on to it (Figure 3.5). The joystick (CH-Hall effect – Appendix A.1) is a robust and low cost device that does not require excessive use of electronics, and its use integrated onto the device reduces the risk of failure (Figure 3.7).



Figure 3.7 The joystick CH-Hall Effect.

The base is supported by two load springs (Figure 3.8, appendix A.1), in order to make the base more manoeuvrable (Figure 3.5). Their position was chosen by placing them along the upper base evaluating the necessary effort to move the base, the comfort and effectiveness to move the joystick. Additionally, it was necessary to take into account the hardness of the springs. On one hand, they cannot be very flexible and malleable, so that the base is not too sensitive and loose, thus transmitting a state of insecurity to the user. On the other hand, it cannot be too hard, or would turn the base to be less sensitive to the users' intentions and require a lot of effort by the user, taking him, eventually, to fatigue. The choice of the springs as well as its position was made empirically, by trial and error.



Figure 3.8 Load Springs

~To extract the signals from the joystick and then study them it was performed an user study with healthy volunteers, with no history of any dysfunction on either upper or lower limbs. These volunteers had to perform simple tasks like moving forward and then turn left or right. It is noteworthy that these tests were performed without any motorized system. This enables to evaluate the real interaction between the user and the upper base support of the walker without the interference of any control strategy. These signals enable to characterize the interaction between both agents.

The joystick outputs three different signals (X,Y,Z), measured in Volts that specify the imposed movement described on the XYZ-axis attached to the joystick. These axes are depicted in figure 3.9. However in this work it is just used two of these signals, Y and Z.

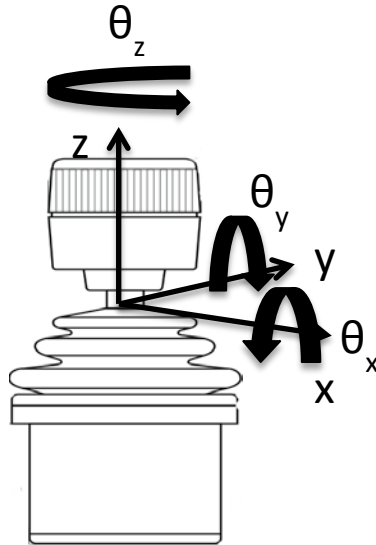


Figure 3.9 Schematic of the joysticks' displacement.

These signals are measured in Volts and are proportional to the rotation angle (θ , $i=y,z$) of the joystick around the Y and Z axes of the joystick that results from the force that is applied by the user on the handles. Signal Y is the angular rotation around the Y-axis and represents the straight or forward movement intention of the user. Signal Z is the angular rotation around the Z-axis and represents the curvilinear movement intention of the user. These are converted by the joystick as follows:

$$\begin{aligned}
 Y &= 0.125 \times \theta_y + 2.5 \text{ (Volts)} \\
 Z &= 0.125 \times \theta_z + 2.5 \text{ (Volts)}
 \end{aligned}
 \tag{3.1}$$

Where θ_y is the rotation angle around the Y-axis and θ_z is the rotation angle around the Z-axis applied to the joystick. Both can vary within the range $[-18^\circ, 18^\circ]$ and are express in degrees. Thus if the joystick is in the rest position the Y and Z are 2.5 V. The value 0.125 is express in Volt per degree.

The acquisition of these signals from the joystick allowed the identification of two main components of the signal. One represents the highest frequency noise caused by the vibrations introduced by the ground and wheels irregularities of the structure, and this component should be eliminated. The other one is the voluntary component that contains the information of the walking movement intentions of the user to guide the walker. This is the one that really interests to guide the walker. These signals and the corresponding processing will be discussed in Chapter 4.

3.3.3.2. *User's gestures while guiding the Smart Walker*

As it was told the directional intent will be identified from physical manipulation of the upper base of the walker. This manipulation is depicted in figure 3.10.

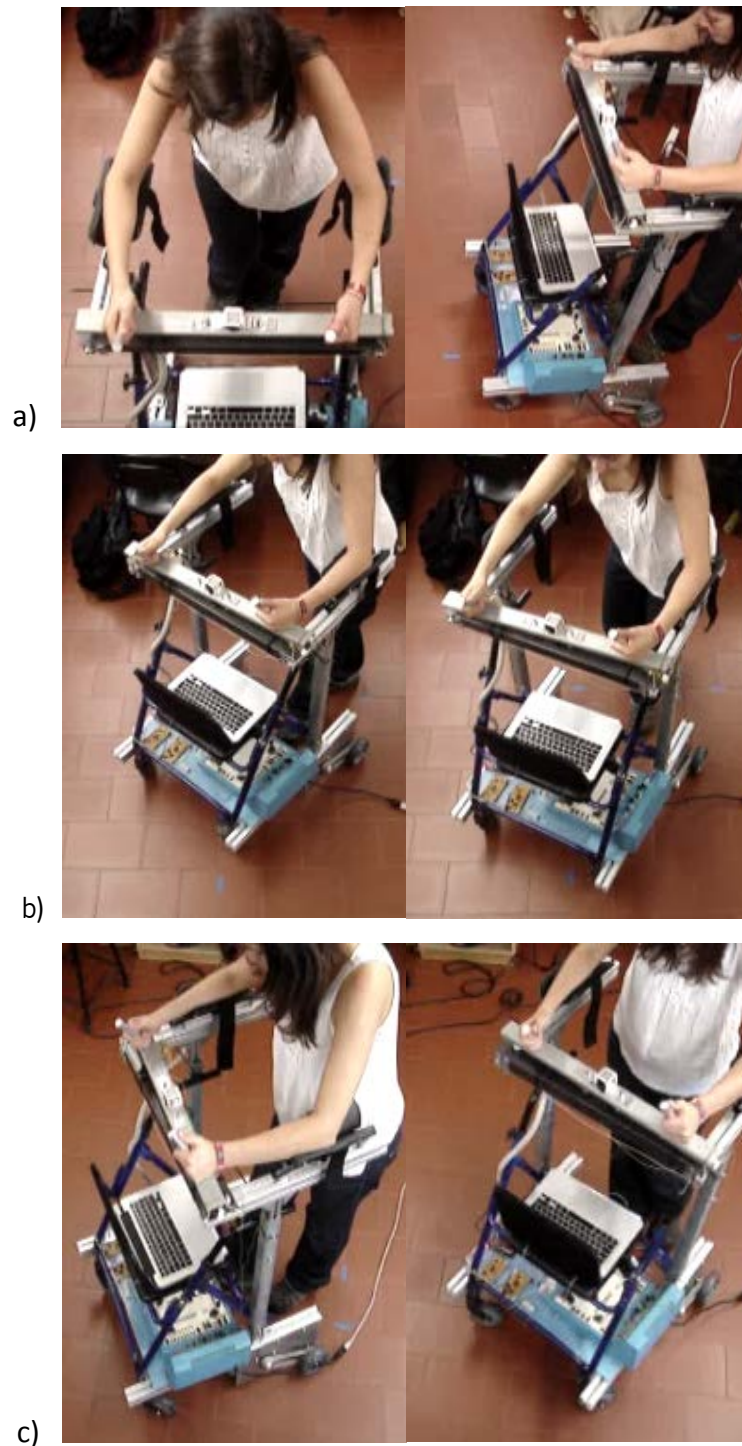


Figure 3.10 The user is a) walking forward, b) turning left and c) turning right.

3.3.4. Fall and Grasp Detection – Safety considerations

The possibility to detect possible falls of the user was one of the concerns and aims integrated on the ASBGo project. Additional safety is envisaged by several sensorial subsystems that complement each other. A motion analysis system has been used to analyse the position of the lower limbs relatively to the walker in order to determine the region of space in which the lower limbs are always present when the user is walking with the walker. A sensor will be put there to detect that area (Figure 3.11a) If the user leaves this area it means that he is possibly falling. Another option to detect possible falls of the user is by monitoring the approximation of the user with infrared sensing at the height of the chest (Figure 3.11b). If the user is falling forwards, the distance between the user's chest and the walker will decrease. The proposed subsystem is composed by infrared sensors and will improve the detection of falling forwards situations during gait and stop the walker in time accordingly if required.

To detect if the user is falling backwards three steps were introduced. Firstly, the walker cannot move backward. So, if the user pushes the upper structure in his direction, the walker stops. Another subsystem ensures the user is guiding the walker grasping the two handlebars. The proposed safety system is compounded by sensors on each handlebar (Figure 3.11b). If the two handlebars are not grabbed, two situations may occur: if the gait is starting, the walker will not move; if the user is already walking with the device, the walker will immediately stop. A third step is based on two sensors, one on each forearm support, that will verify if the user is with his forearms properly supported on the base supports (Figure 3.11b).

3.3.5. Modular Architecture for Data Acquisition and Processing

The proposed sensorial subsystems have to be integrated onto an electronic architecture with communication possibilities, able to do data acquisition, processing and control in real time.

3.3.5.1. NeoASAS Architecture

For this purpose, in the NeoASAS walker, MATLAB Real-Time Workshop was used, which allows the creation of real-time systems from SIMULINK blocks. A full application, including the program

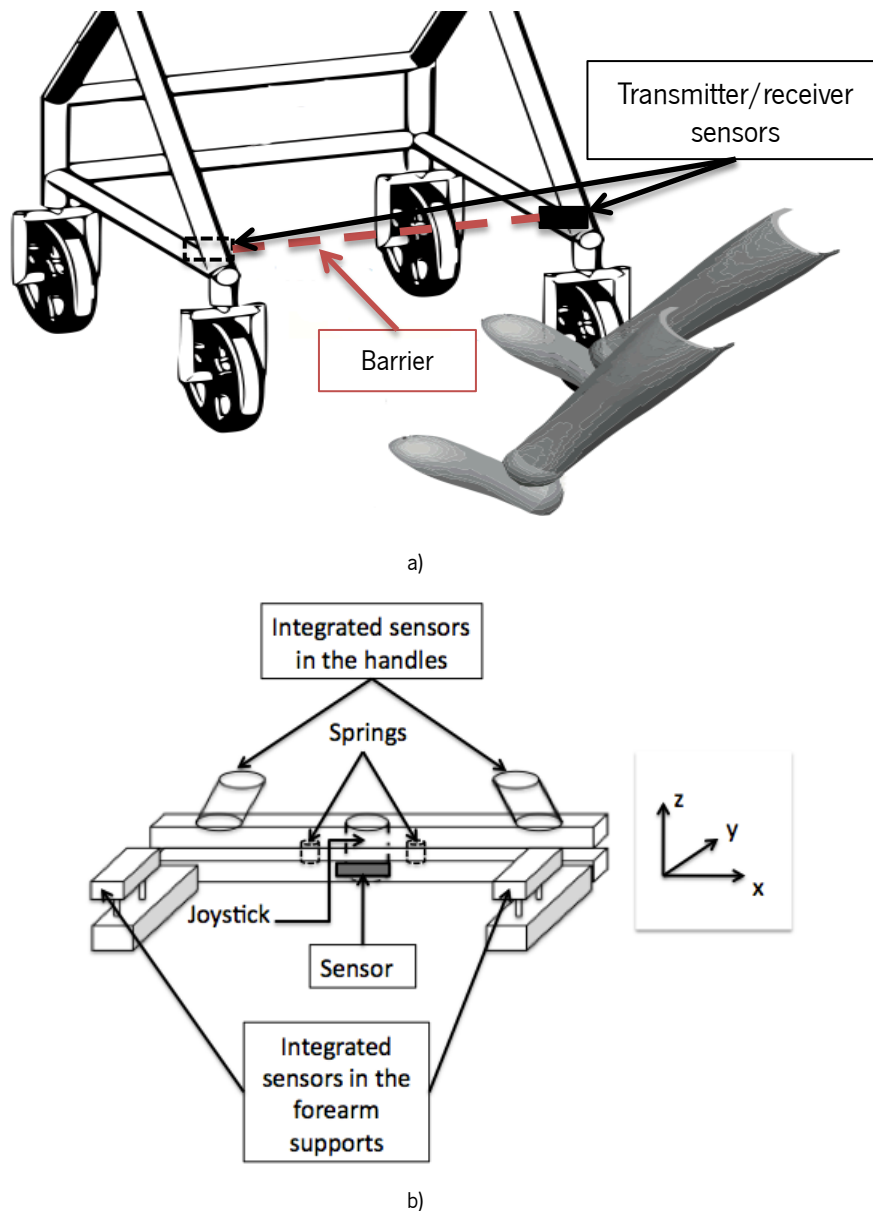


Figure 3.11 Schematic of the sensors integrated in a) the lower base and b) the upper base of the walker.

that will be executed on real-time, can be developed in short time, taking the advantage of using all analysis and data processing tools of MATLAB.

In CSIC, the processing elements used were:

- Conventional PC portable, ThinkPad Z61m of Lenovo.
- CPU of Seco, M585R model. This one belongs to a PC/104-Plus platform Intel Celeron 650 MHz, with 128 MB of RAM and 32 MB of FLASH-IDE.

This mixed architecture composed by these two elements plus the MATLAB were chosen because:

- There is an increase of computational capacity of the system, allowing the implementation of complex algorithm in real-time;
- Possibility of using other programs in parallel with the system of real-time executed in the PC/104, allowing a more universal system and increasing the possibilities of communication with other existing systems.
- It enables to develop a system without taking into consideration the programming problem, and focus first on the development and validation of concepts.

There is also the possibility to connect the sensor subsystems incorporated on the walker through an acquisition card (PC/104 Standard) of the Diamond Systems Corporation, DMM-32X-AT model. This card has input/output digital (3 ports with 8 bits) and analog (32 single-ended with 16 bits and auto calibrated) channels.

To program the PC/104-Plus and the communication with the portable PC during the acquisition, it is used a network card from Advantech, PCM-3663 model.

To control de motors it was used a board with a dsPIC304011, which had been programmed to receive data from the PC/104 and then sent to the motors of the walker.

Finally, to power the system it was developed a source powered with a battery of 12V 9Ah, which outputs 5V 16A based on the converter of Texas Instruments PTH08T220W.

3.3.5.2. ASBGo Architecture

At University of Minho, it was chosen other architecture to acquire and process the signals sent by the sensorial systems to control the motors of the ASBGo walker.

The PC is a fundamental element to control, acquire and analyze data, however it incurs high cost.

As a low cost alternative it was selected an Arduino board (Chip Kit Max 32), based on a versatile microcontroller that potentiates his functions beyond a simple passive interface of data acquisition, operating alone on the control of various devices and this having applications in embedded instrumentation and robotics.

Arduino is a hardware open source platform of simple use, ideal for creating an interaction between input sensors (sound, light, etc) and output LED's, motors, etc. The platform has a bootloader and a friendly interface on the computer that uses language Processing [96] based on

C/C++, which is also open source. Through the bootloader it is dispensed the use of programmers for the chip (ATMEL), and requires no additional hardware and compilers. In this development environment, libraries are available and allow interfacing with other hardware, allowing the full development of simple or complex applications in any area.

So, Arduino will substitute the PC/104 and the motor board control. This option relies onto the reduction of the amounts of electronics, lower cost device and focusing the concentration on executing the tasks faster, using a platform where it is possible to develop software in a easy and faster way. Thus, this can be an autonomous and complete platform that receives inputs from the sensorial systems and directly controls the motors. Moreover, it is low cost and low battery consumption. Despite being more robust, the PC/104 consumes a lot of energy, is expensive and requires more electronics, such as the motor board control.

For this propose, to develop a device with a commercial focus, Arduino seems to be a good option for this application.

3.4. Conclusions

In this chapter is done a brief presentation of the ASAS and SIMBIOSIS projects, followed by a presentation of the goals of the NeoASAS and ASBGo walkers.

Therefore, the NEOASAS and ASBGo projects emerged in order to give continuity to the ASAS and SIMBIOSIS projects, but with a commercial focus.

The NeoASAS and ASBGo aim to provide for good support through the use of the forearm supports and stability by placing the heavy elements on the lower base of the walker. These projects also aim to try to offer a certain degree of intelligence in assistance and judgement. Information gathered by several sensors is used to inform the Smart Walker about the status of the user, i.e. identify the movement intentions of the user considering direction and velocity through a joystick placed onto the upper base of the walkers, as well as additional safety with infrared and force sensors, by detecting possible falls of the user.

4. Validation of the Proposed Interface

4.1. Introduction

To qualitatively evaluate the information and sensibility that the joystick, as an interface, can provide regarding user's command intentions, a series of experiments were performed. The joystick was mounted onto the walker as previously described in Chapter 3. An acquisition platform was built to acquire and process the signals obtained from the joystick. Experiments were performed without any motorization in order to evaluate the real interaction between the user and the walker without the interference of any control strategy. Experiments consisted on three different pre-established trajectories with five individuals. The purpose was to obtain a qualitative relationship between the performed trajectories and the signals sent by the joystick.

A comparison among these showed that the amplitude of the joystick signal depends on the rotation of the joystick due to the force applied by the user on the handles when walking and on the intention to perform the pre-established trajectory. So, it was experimentally observed that the joystick, as an interface, presents enough sensibility to make a correct and effective interpretation of the users' commands.

It was also noticed that the acquired signals have two types of components: one that contains the user's command intentions and other a higher frequency components. This last one may interfere in the readings of the joystick, and cause a misinterpretation, and therefore has to be eliminated by a filter strategy.

The performed experiments and the achieved results with the NeoASAS and ASBGo joystick's signals are presented and discussed in this section. An introduction and comparison between filter strategies to estimate the user commands on the joystick signals, eliminating the higher frequency components, are presented for each walker, as well as the performance of these filters, which serve to define an optimal filter strategy to this application.

4.2. Evaluation of the signal related with the users' command intentions in the NeoASAS and ASBGo walkers.

To begin with the study of the filter and control strategies for the motorized movement of the NEOASAS and ASBGo walkers, it is first necessary to perform some experiments with the user walking with the walker, and then analyse the joystick signals that are obtained.

These experiments were executed without any motorization in order to enable the evaluation of the real interaction between the user and the walker without interference of any control strategy. Thus, three different trajectories were established with five subjects, and each was repeated five times. The three trajectories are presented on table 4.1.

Table 4.1 Description of the three trajectories (A, B and C) performed by five individuals, five times each.

Trajectory	Steps	Description	Scheme
A	S1	Stopped.	
	S2	Start walking forward.	
	S3	Turn Right 90°.	
	S4	Go forward.	
	S5	Stop.	
B	S1	Stopped.	
	S2	Start walking forward.	
	S3	Turn Left 90°.	
	S4	Go forward.	
	S5	Stop.	
C	S1	Stopped.	
	S2	Start walking forward.	
	S3	Stop.	
	S4	Start walking forward.	
	S5	Stop.	

The duration of the experiments depend on the subject and his velocity, but took around 25-30 seconds.

On each experiment two types of signals are acquired and evaluated – forward (Y) and rotation (Z). The rotation around the Y-axis of the joystick (Figure 3.9), measured by Y-signal, gives an indication of the user intention to move forward. According to the applied force on this axis, the signal will have more or less amplitude, depending on the user's command intention to go forward with more or less velocity. The rotation around the Z-axis of the joystick (Figure 3.9), measured by Z-signal, gives an indication of the user intention to perform a curve. The signal will present high or low amplitude depending if the performed curve is more or less accentuated. The intention to turn right or left is detected by the sign of the signal, i.e. turn left causes negative signal and turn right causes positive signal.

To study the signals it was taken the offset (2.5V) of the amplitude of the Y and Z-signals, making them to oscillate around zero.

Figure 4.1 show typical Y and Z joystick data, corresponding to the three different trajectories (Table 4.1) performed with the NeoASAS walker.

Figure 4.2 present typical Y and Z raw joystick data, obtained when the user was performing the three different trajectories (Table 4.1) with the ASBGo walker.

Comparing the signals of the two walkers, for the corresponding trajectories, one can see that the two integrated joystick systems read the performed trajectories in the same way. The signals are similar, as it was expected.

To analyse the pre-established trajectories represented by the joystick signals (Y and Z), the ensemble of the two figures was made and is illustrated in figure 4.3.

First, it is illustrated, in figure 4.3, the two signals, Y and Z, acquired during trajectory B. The different steps performed during the B trajectory are identified in the figure by letters S_i, i=1,2,3,4,5.

Initially the user is stopped (S1) and both Y and Z signals are zero. When the user begins to walk forward (S2), he pushes forward the handles (Figure 3.5) of the walker and the Y-signal becomes negative, because the joystick is moved around the Y-axis (Figure 3.9). The Z-signal stays zero, since the joystick is not rotated around the Z-axis when the user walks forward. Z-signal may present small variations when the user is moving forward, caused by asymmetrical efforts applied by the user. This can cause the joystick signal to present a slight variation towards one direction of the Z-axis.

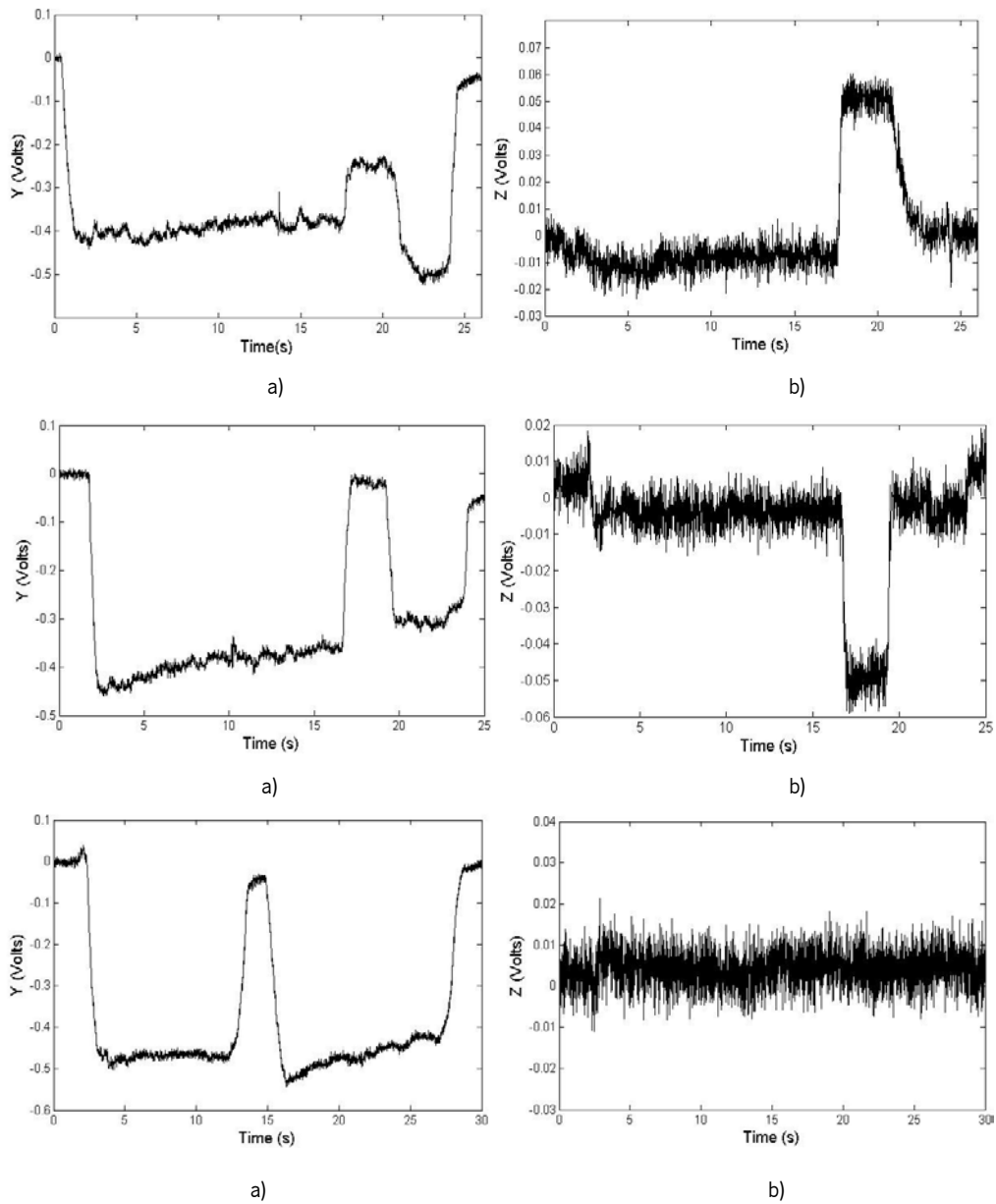


Figure 4.1 Typical raw Y (a) and Z (b) joystick data in the NeoASAS walker when the user is performing the three different trajectories explained in table 1: Top) A, Middle) B, Bottom) C trajectories.

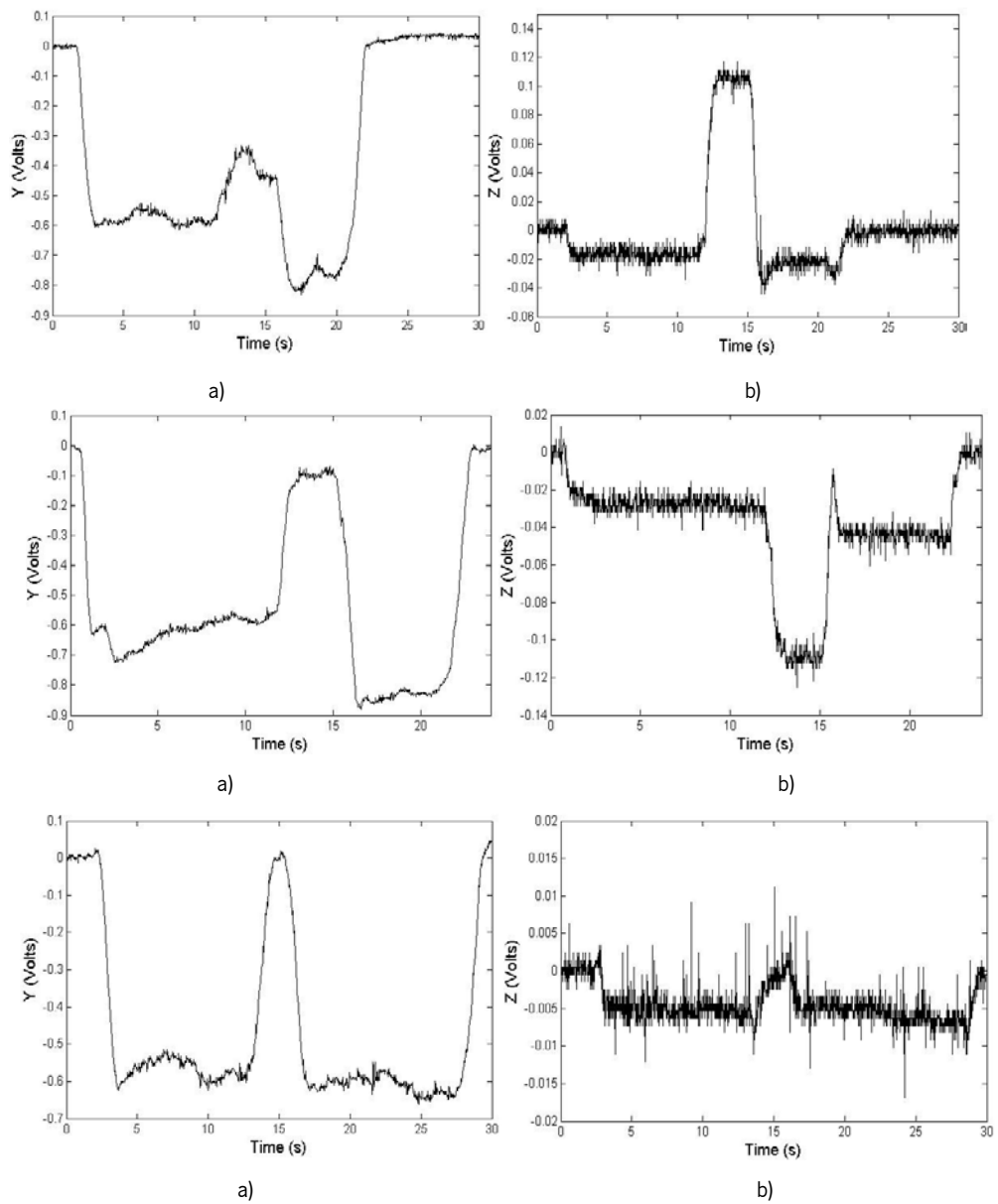


Figure 4.2 Typical raw Y (a) and Z (b) joystick data in the ASBGo walker when the user is performing the three different trajectories explained in table 1: Top) A, Middle) B, Bottom) C trajectories.

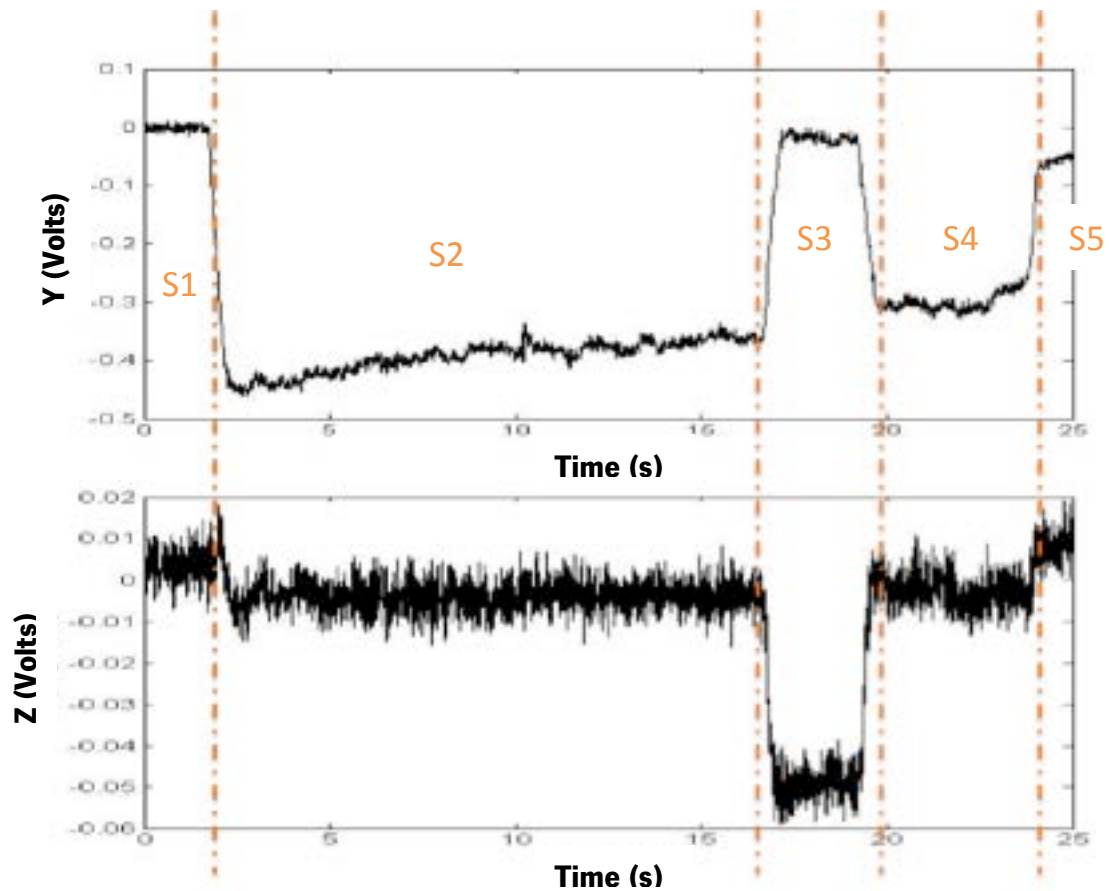


Figure 4.3 Different phases of the trajectory B. S1- The user is stopped; S2- User starts walking forward; S3- User turns right; S4- User walks forward; S5 – User stops.

When the user turns right (S3), the joystick is made to move around the Z-axis to the right side also, presenting a negative Z-signal. At the same time, in the Y-axis the joystick tends to go to rest, presenting a zero Y-signal. This happens because the user wants to make a tight curve, making the walker to rotate 90° , approximately. This is called a rotational curve, where no forward movement is made. On the other hand, when the user wants to make a slight curve the joystick doesn't go to rest in the Y-axis, because the user continues to want to go forward, while turning to a side. This is called a translation curve. An example of a translation curve performed in a B trajectory is depicted in figure 4.4. A translation curve is performed during the time in between the two dashed lines, $t = [15, 20]$ s. During this time, the Y-amplitude almost does not change but it is not zero. However, the Z-signal detects the rotational intent as expected

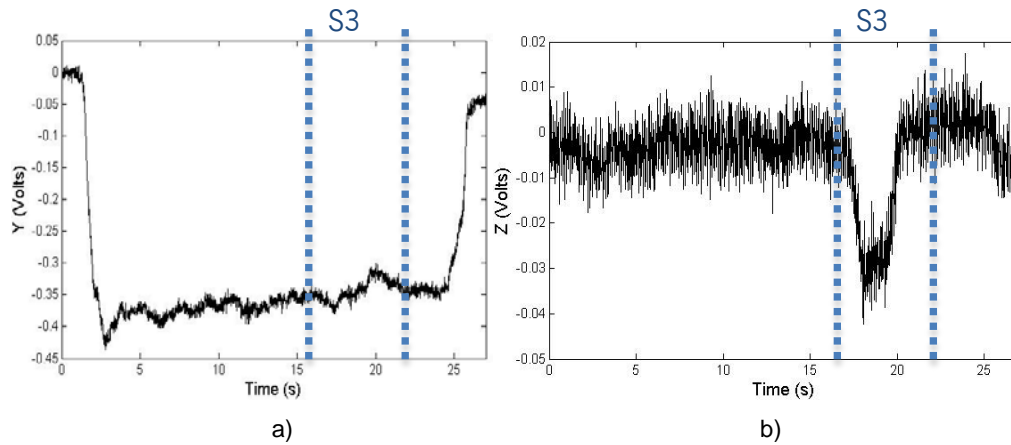


Figure 4.4 Typical raw Joystick data a) Y-signal and b) Z-signal obtained when the user is performing a translation curve (trajectory B) with the ASBGo walker.

While performing trajectory B, the next step is to move forward (S4), and the Z-signal returns to zero while the Y-signal becomes negative (Figure 4.3). At the end of the trajectory the user stops (S5), and the Y-signal returns to zero. Z-signal remains in zero. This happens when the user wants to stop, pushing to himself the handles of the walker.

The analysis of trajectory A is analogous to trajectory B. The only difference is that the Z-signal, in a curve, becomes positive instead negative.

In trajectory C (Figure 4.5), steps (S1) and (S2) are analogous to steps (S1) and (S2) in trajectory B. However, step (S3) is to stop and the Y-signal returns to zero. Steps (S4) and (S5) are analogous to steps (S1) and (S2), where the user begins to walk forward, and then stops, pushing the handles to himself. Let it be considered the Y-signal depicted in top figure 4.2, relative to trajectory A when performed with the ASBGo walker. The Y-signal is positive from $t = [28, 30]$ s. As it will be discussed later, this situation depicts a dangerous situation to be detected.

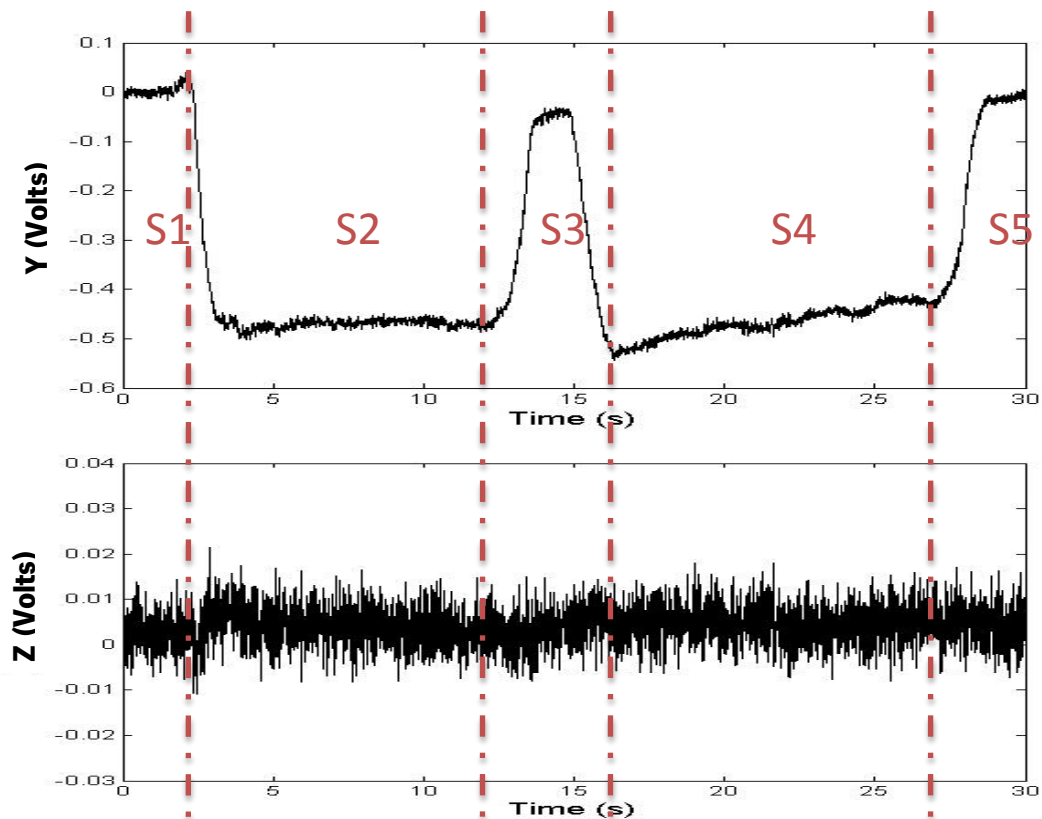


Figure 4.5 Different phases of the trajectory C. S1- The user is stopped; S2- User starts walking forward; S3- User stops; S4- User walks forward; S5 – User stops.

As it was expected the Z signal is always zero since the user doesn't want to turn aside.

After this discussion, one can conclude that the joystick system read correctly the user's command intentions.

Observing now the Z and Y signals, two main components of the signals: (1) One represents the highest frequency noise caused by the vibrations of the structure due the irregularities of the ground; electromagnetic noise and wheel eccentricities. This component must be eliminated in real-time, without causing a considerable delay on the signal. For that, it will be used a tracking filter and the choice will be presented in detail on the next section.

The other component (2) is the voluntary component that contains the information relative to the walking movement intentions of the user to guide the walker. This is the one that really interests to guide the walker. It enables to characterize the trajectory that the user pretends to follow, making possible the continuous monitoring of the users' movement, allowing the development of robust and secure control strategies.

4.3. Joystick's Signal Processing Architecture

For each walker, an architecture to process the acquired joystick signals was developed.

In the NeoASAS platform, the signal acquisition consisted in sending the raw Y and Z signals through PC/104 to a signal processing architecture based on Matlab Real-Time xPC Target Toolbox with a sampling period of 1 ms, as it was presented in Chapter 3. The architecture, constructed in Matlab, Simulink, consisted in three blocks: Offset, Filter Strategy and Amplification (Figure 4.6).

The offset block eliminates the offset from the Y and Z-signals, making them to oscillate around zero, Y_o and Z_o . These signals Y_o and Z_o are filtered in the filter strategy block using a filter which choice is made in section 4.4.1, resulting in Y_f and Z_f . Each of these signals is afterwards multiplied by a different gain in the Amplification block resulting in Y_g and Z_g .



Figure 4.6 Diagram of the processing architecture of the Y and Z-signals.

In the ASBGo walker, it was used the ChipKit Max 32 (Arduino platform) to acquire the signals with a 0.01s of sampling period, as it was presented in Chapter 3. This hardware has a voltage limit of 3.3V, in contrast to the maximum voltage of 5V of the joystick, and a resolution of 3 mV. The three blocks in Figure4.6 were implemented in hardware circuit. If necessary, the signals are then filtered by a digital filter in the Arduino platform.

These architectures will be explained in more detail in the next sections.

4.4. Signal processing architecture of the NeoASAS walker

The Signals processing architecture blocks implemented in the NeoASAS were:

i) The Offset block

The Offset block of the architecture, (Figure 4.6), eliminates the 2.5V offset from the input Y and Z-signals. In order to do so, the signals are subtracted with the mean of the first 1000 signal samples, which corresponds to 1s of sampling. The resultant signals oscillate around zero as

shown in figures 4.1 and 4.2.

ii) Filter Strategy

The Filter Strategy block aims to eliminate in real-time the noise (high-frequencies) components present in the acquired signals. The higher-frequency components are related to noise caused by structure vibrations, the irregularities of the ground, electromagnetic noise and wheel eccentricities.

The output of this block is the signal used to control the device's movement, converting user's command intentions into motor inputs.

This specific block will be presented and discussed in detailed in the next section.

iii) Amplification block

The Amplification block will amplify the Y and Z-signals, since their amplitudes are very small. The gain chose to amplify the Y signal is equal to -3 and for the Z-signal is 30. These gains were chosen to amplify the signals to the range [0,1] for the Y-signal and [-1,1] for the Z-signal to then be used on a control strategy to control the motors of the walker, as it is going to be seen in Chapter 6.

4.4.1. Selection and implementation of a filter strategy

The data that was collected yields that the user's commands intentions occur in a frequency range between 0 and 2 Hz in both Y and Z-signals. In figure 4.7, one can see the spectrum for two typical Y and Z-signals.

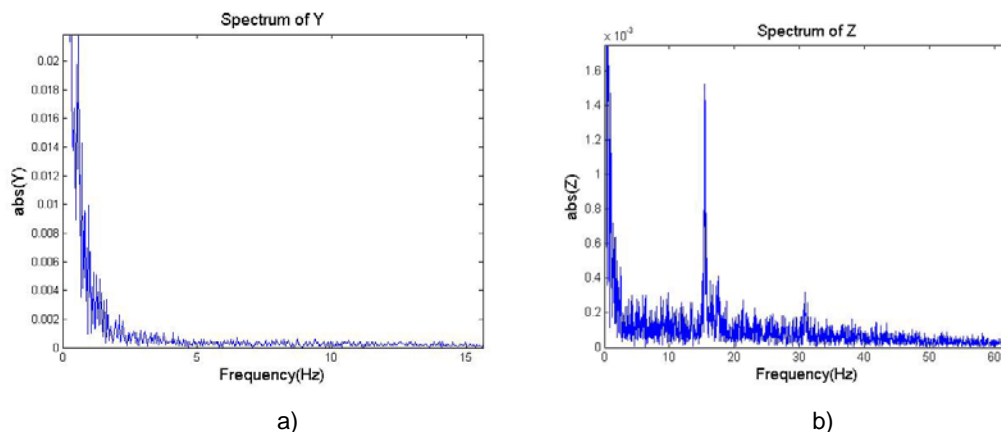


Figure 4.7 Example of a Frequency Spectrum for a) raw Y-signal and b) raw Z-signal. It is observed that user's command intentions are below 2Hz and have considerable more energy than the noise components (higher frequencies).

The Z-signal has more accentuated higher-frequencies than the Y-signal, mainly between 15-20 Hz. This accentuation can be evaluated with the calculus of the signal-to-noise ratio (SNR) [97]:

$$SNR = \frac{\mu}{\sigma} \quad (4.1)$$

where μ is the signal mean and σ is the standard deviation of the noise.

The SNR was calculated for all the acquired data and then it was calculated the SNR mean for each signal.

Table 4.2 presents the SNR average value for each signal. The SNR of the Z-signal is much lower than the one of the Y-signal. This means that Z-signal has to be further filtered comparing with the Y-signal.

Table 4.2 SNR (signal-to-noise ratio) of the Y and Z signals (raw and filtered with the Butterworth filter).

Signal	Y	Y(filt)	Z	Z(filt)
SNR	48.98	59.72	2.37	9.95

The higher frequency components present in the signals can be eliminated with forth and back recursive digital filters, such as Butterworth filters, without causing phase distortion [98]. This technique consists in filtering input data in both the forward and reverse directions: after filtering in the forward direction, the algorithm reverses the filtered sequence and runs it back through the filter, which yields precisely zero-phase distortion. However, this approach is not real-time implementable. In figure 4.9, it is shown the result of the filtering of the Y and Z-signals with a cut-off frequency of 2 Hz.

Observing these results, one can conclude that this filter eliminates/attenuates with efficiency the higher frequency components (above 2 Hz), having a higher SNR (Table 4.2). As this technique is not suitable for real-time applications, this results show that this filter is a good reference for an ideal signal to be achieved by the filter strategies to be chosen. Thus, this filter is set as a basis to evaluate the performance of the chosen filter strategy.

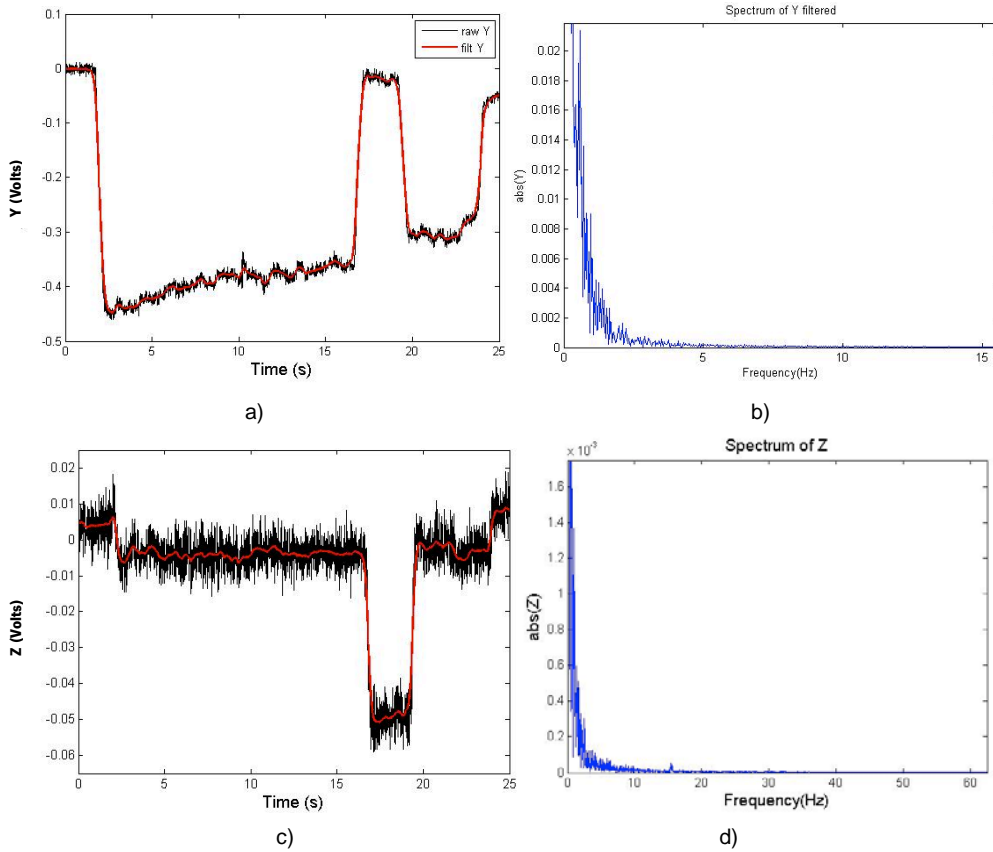


Figure 4.8 Attenuation of higher frequency components by means of recursive digital filters (Butterworth) with 2Hz of cut-off frequency. a) The raw Y-signal (black) and the filtered Y-signal (red) with the digital filter; b) and d) Frequency Spectrum of Y and Z filtered, respectively; c) The raw Z-signal (black) and the filtered Y-signal (red) with the digital filter.

As it was discussed above, the user's command intentions are performed between 0 and 2Hz. Thus, the filter will be designed to neglect components above 2 Hz.

Considering equation (4.2), which represents a Taylor series, the second derivative can be neglected if either the sampling period T_s or the acceleration itself are small [99]. Since the second derivative maximum average value is 4.16×10^{-10} rad/s³ for the Y-signal and 2.48×10^9 rad/s³ for the Z-signal, and the sampling period T_s is 1ms, one can neglect the second derivative ($\ddot{x} = 0$) and all the higher order.

$$x(t) = x(t_n) + T\dot{x}(t_n) + \frac{T^2}{2}\ddot{x}(t_n) + \dots \quad (4.2)$$

Thus, the user's command intentions can be modelled as a first order process.

The filter to be chosen has to be suitable to be implemented in real-time. Additionally, the filter will have to estimate the user's command intentions, neglecting the noise. A low computational cost is needed to not require expensive hardware to process the signal and it should not cause a

considerable temporal delay on the filtered signal, because in the real application when the signal is used to control the walker, the user should not perceive the delay between his commands and the movement of the walker. This human perception is around 200 ms [100].

In the literature [98-99] [101-102], two types of filters were identified to be frequently used as potential candidate algorithms to eliminate components of higher-frequency in real-time. These filters are the g-h filter (Benedict-Bordner and Critically Damped) and the Kalman filter, and they are usually called as tracking filters.

i) **g-h Filter**

g-h filters are simple recursive filters that estimate future values of the signal based on a first order model of the process. Measurements are used to correct these estimations/predictions, minimizing the estimation error [99].

The follow derivation comes from [99].

Let x_k and \dot{x}_k denote a state and its derivative at time step k, and x_{k+1} and \dot{x}_{k+1} denote a state and its derivative at time step k+1.

$$x_{k+1} = x_k + \dot{x}_k T \quad (4.3)$$

$$\dot{x}_{k+1} = \dot{x}_k \quad (4.4)$$

where T is the time between time steps.

These equations are called the *system dynamic model*.

Suppose at time k, a measurement y_k is different from the estimated state of x_k based on past measurements ($\hat{x}_{k,k-1}$). This requires an update of the first derivative state \dot{x}_k , as follows:

$$\hat{\dot{x}}_{k,k} = \hat{\dot{x}}_{k,k-1} + \frac{h_k}{T_s} (y_k - \hat{x}_{k,k-1}) \quad (4.5)$$

where T_s is the sampling period. h_k is a small parameter that determines the fraction of the measurement error applied to the first derivative prediction.

$\hat{\dot{x}}_{k,k-1}$ is the *predicted estimate*, i.e. the estimation of \dot{x}_k at time step k based on past measurements. $\hat{\dot{x}}_{k,k}$ is the *filtered estimate*, i.e. the estimation of \dot{x}_k at time step k based on past measurements as well as on the use of the current measurement y_k .

The estimated state can be updated in a similar way:

$$\hat{x}_{k,k} = \hat{x}_{k,k-1} + g_k(y_k - \hat{x}_{k,k-1}) \quad (4.6)$$

where g_k is a small parameter that determines the fraction of the measurement error applied to the state prediction.

Taken together, the two estimation equations form the *g-h track update* or *filtering equations*:

$$\hat{\dot{x}}_{k,k} = \hat{\dot{x}}_{k,k-1} + \frac{h_k}{T_s}(y_k - \hat{x}_{k,k-1}) \quad (4.7)$$

$$\hat{x}_{k,k} = \hat{x}_{k,k-1} + g_k(y_k - \hat{x}_{k,k-1}) \quad (4.8)$$

Now, the *system dynamic* equations can be used to predict the state and its derivative at time step $k+1$, according to equations (4.3) and (4.4).

First, equations (4.3) and (4.4) have to be rewritten using the new notation to obtain the *g-h state transition* or *prediction equations*:

$$\hat{\dot{x}}_{k+1,k} = \hat{\dot{x}}_{k,k} \quad (4.9)$$

$$\begin{aligned} \hat{x}_{k+1,k} &= \hat{x}_{k,k} + \hat{\dot{x}}_{k,k} T_s \\ &= \hat{x}_{k,k} + \hat{\dot{x}}_{k+1,k} T_s \end{aligned} \quad (4.10)$$

In equation (4.10) the estimated first derivative at time $k+1$, $\hat{\dot{x}}_{k+1,k}$, is equal to the value $\hat{\dot{x}}_{k,k}$ at time k , because a constant first derivative model is assumed. In this application this can be assumed considering the human movements are slow. This assumption is reasonable as long as the time between observations T_s (sampling period) is small or the second derivative \ddot{x} is small or their combination is small [99]. As it was seen previously in this section, this application follows these requirements.

The equations (4.9) and (4.10) allow the transition from the estimated state and its estimated first derivative at time k , $\hat{x}_{k,k}$ and $\hat{\dot{x}}_{k,k}$, to the estimated state and its estimated first derivative at

time $k+1$, $\hat{x}_{k+1,k}$ and $\hat{\dot{x}}_{k+1,k}$. So they provide a prediction of future state and its first derivative, $\hat{x}_{k+1,k}$ and $\hat{\dot{x}}_{k+1,k}$, based on a *first order dynamic model* of the system.

The specific equations to update and predict the state and its first derivative are present below in table 4.3.

Table 4.3 Update and Predictions equations of the g-h filter

Update Equations	Prediction Equations
$\hat{x}_{k,k} = \hat{x}_{k,k-1} + \frac{h_k}{T_s} (y_k - \hat{x}_{k,k-1})$	$\hat{x}_{k+1,k} = \hat{x}_{k,k}$
$\hat{\dot{x}}_{k,k} = \hat{\dot{x}}_{k,k-1} + g_k (y_k - \hat{x}_{k,k-1})$	$\hat{\dot{x}}_{k+1,k} = \hat{\dot{x}}_{k,k} + \hat{\dot{x}}_{k,k} T_s$

These equations have two parameters that have to be tuned prior to operation of filter, which are g_k and h_k .

These parameters can be constant, making the computations required by the filter to be very simple and the memory requirements very small. Specifically, for each y_k only two storage variables are required, one for the latest predicted first derivative $\hat{\dot{x}}_{k,k}$ and one for the latest predicted state. Past measurements and past predicted states are not needed for future predictions.

g-h filters are affected by two error sources [99]: systematic error, which is related to the constant first derivative assumption and the measurement error, which is inherent to the sensor and measurement process.

Typically, the smaller g_k and h_k are, the larger the systematic error and the smaller are the measurement error [99]. The selection of these parameters determines whether we put the combined estimate closer to y_k or to $\hat{x}_{k,k-1}$. If $\hat{x}_{k,k-1}$ is much more accurate than y_k , then the parameter g_k will be smaller, so that the filter trusts less in the measurement than in the predicted state [99]. Therefore, when designing a g-h filter there is a degree of freedom in the choice of the relative magnitude of the measurement and systematic errors.

To simplify the selection of parameters, it will be considered two filters that consider the g and h as constant. These filters are the Benedict-Bordner Filter [103] and the Critically Dampened Filter [99], described next.

Benedict-Bordner Filter (BBF) minimizes the total transient error $D_{x_{k+1,k}}$, defined as the sum of the squares of the difference between the true state and the predicted state (4.11). In the literature [99] [103] such a minimization was carried out for the weighted (λ) sum of $D_{x_{k+1,k}}$ plus the variance of the prediction error due to measurement noise errors (4.12):

$$D_{x_{k+1,k}} = \sum_{k=0}^{\infty} (x_{k+1,k} - x_{k+1})^2 \quad (4.11)$$

$$E = \text{var}(x_{k,k-1}) + \lambda D_{x_{k+1,k}} \quad (4.12)$$

According to the mathematical formulations found at [99], the BBF relates g and h by:

$$h = \frac{g^2}{2 - g} \quad (4.13)$$

With this formulation, there is just one degree of freedom for the parameters choice.

In figure 4.9, one can see the influence that the choice of the g parameter has on the performance of the BBF filter.

If $g=0.5$ (Figure 4.9a), the filtered signal is equal to the raw signal. This indicates that the estimated and the measured signal have the same accuracy, and thus they are equal. Then, one can see that if g diminishes, the filter trusts less and less on the measurements, and starts to filter the signal, approximating to the ideal Z . In $g=0.001$ (Figure 4.9c) the filtered signal is starting to have a significant delay, which means that the filter is not trusting in the measurements, so it takes more time to adjust to the signal.

With this analysis, one can conclude that the choice of the g parameter have to take into consideration that the high frequency components (noise) have to be eliminated, without introducing delay on the signal. The value of g will depend on the SNR of a signal. If a signal A has a small SNR (very noisy) when compared with a signal B, g should be lower for the signal A than for the signal B, because it has to be further filtered. Meaning that the filter should believe more in the estimates than in the measures.

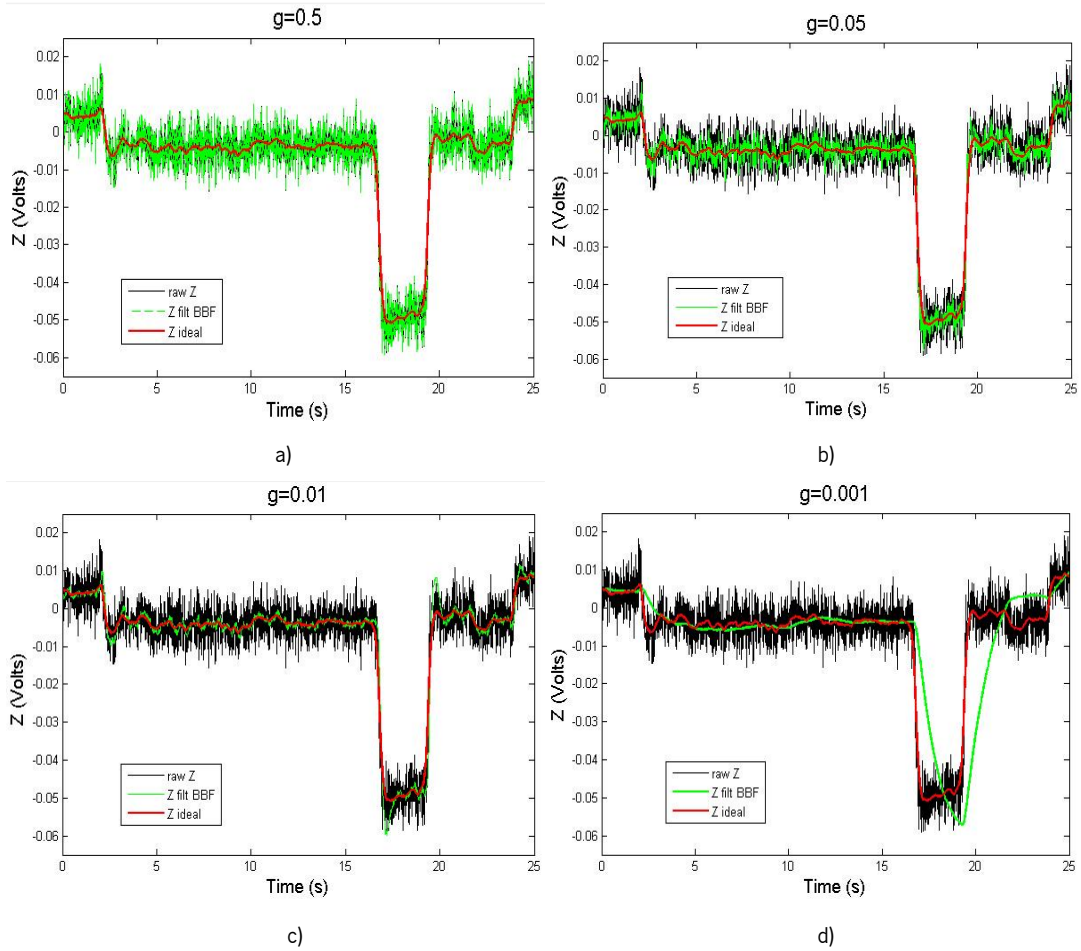


Figure 4.9 Influence of the variation of the g parameter on the filter BBF. a) $g=0.5$, b) $g=0.05$, c) $g=0.01$, d) $g=0.001$. All graphs represent the raw Z signal in black, the filtered Z signal with BBF in green and the ideal Z in red.

Critically Dampened Filter (CDF) minimizes the least squares fitting line of previous measurements [99]. The least square fitting line is the line that gives the minimum error between the line and the measurements. So, the intuit is to minimize the sum of squares of the difference between the measurements and the straight-line fit. The deviation between the straight line at time k and the measurement, at time k , is given by:

$$\mathcal{E}_k = x_k - y_k \quad (4.14)$$

where x_k is the position range indicated by the straight-line fit at time k . Then the least-squares fitting line is the line that minimizes the total sum of squares of differences given by \mathcal{E}_T :

$$\mathcal{E}_T = \sum_{k=0}^K \mathcal{E}_k^2 \quad (4.15)$$

where K is the number of measurements.

This approach weights the old data equally as importantly as the latest data when calculating the least squares fitting line that predicts where the signal would be at the next measurement. However in the CDF approach, old data will have a reduced influence when predicting the future values of the signal. This is done by weighting the older errors, ε_k^2 in equation (4.16), with a factor, θ , less than 1 when forming the total error sum, ε_D :

$$\varepsilon_D = \sum_{r=0}^K \theta^r \varepsilon_{(K-r)}^2 \quad (4.16)$$

According to the mathematical formulation, presented in [99], the CDF relates the parameters g and h to θ by:

$$\begin{aligned} g &= 1 - \theta^2 \\ h &= (1 - \theta)^2 \end{aligned} \quad (4.17)$$

In figure 4.10, one can see the influence that the choice of the θ parameter has on the performance of the CDF filter. If $\theta=0.5$ (Figure 4.10a), the filtered signal is equal to the raw signal. This indicates that the estimated and the measured signal have the same accuracy, and thus they are equal. Then, one can see that if θ increases, the filter trusts less and less on the measurements, and starts to filter the signal. In $\theta=0.998$ (Figure 4.10c) the filtered signal is starting to have a significant delay, which means that the filter is not trusting in the measurement, so it takes more time to adjust to the signal.

With this analysis, one can conclude that the choice of the θ parameter has to take into consideration that the high frequency components (noise) have to be eliminated, without introducing delay on the signal. It can also be conclude that the value of θ depends of the SNR of the signal that is going to be filtered. So, in this case, if a signal A has a high SNR when compared with a signal B, θ should be higher for the signal A than for the signal B, giving more importance to the older errors of signal A. Consequently, g parameter should be lower and the filter will believe more in the estimates than in the measures.

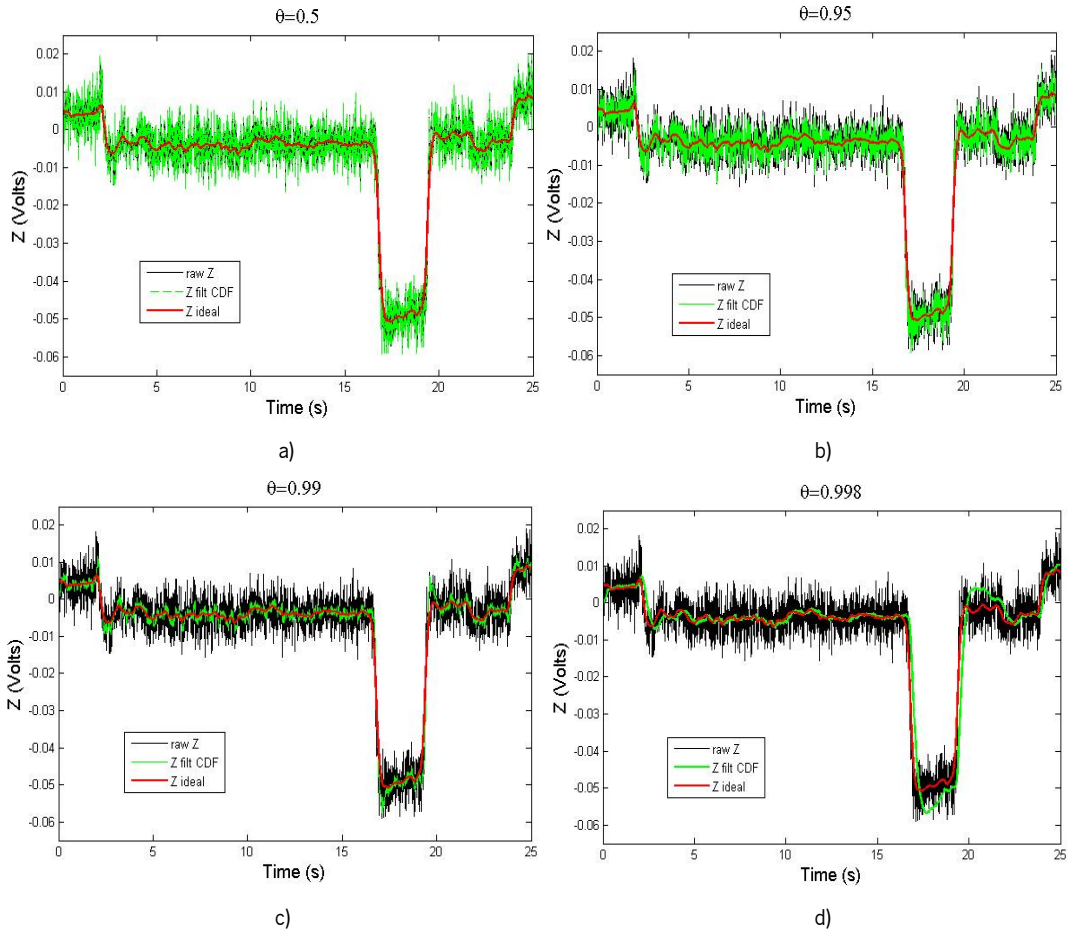


Figure 4.10 Influence of the variation of the θ parameter on the filter CDF. a) $\theta=0.5$, b) $\theta=0.95$, c) $\theta=0.99$, d) $\theta=0.998$. All graphs represent the raw Z signal in black, the filtered Z signal with CDF in green and the ideal Z in red.

ii) Kalman Filter

The Kalman filter has been subject of extensive research [98] [101-102] and application being the most widespread estimation algorithm. This filter provides an efficient computational tool to estimate the state of a process, in a way that minimizes the mean of the squared error [104]. The Kalman version here depicted is the conventional Kalman filter and is only suitable for linear systems.

The Kalman filter addresses the general problem of trying to estimate the state vector $X_{k,k-1}$ of a process that is governed by the linear stochastic difference equation (4.18) [99] [102], designated by State equation:

$$X_{k,k-1} = AX_{k-1,k-1} + W_{k-1,k-1} \quad (4.18)$$

where k is the current instant of time. $X_{k-1,k-1}$ is the state vector, $n \times 1$ dimensions, and A is a state transition matrix, with $n \times n$ dimensions, where n is the number of states. It relates the state at the previous time step $k-1$ to the state at the current time step k , in the absence of process noise. This matrix can change with time, but here it will be assumed constant.

The Observation equation (4.19) with a measurement Y_k is represented by :

$$Y_k = HX_{k,k-1} + V_{k,k-1} \quad (4.19)$$

Y_k is the measurement vector and H is the observation matrix, with $r \times n$ dimensions, where r is the number of outputs. It relates the state $X_{k,k-1}$ to the measurement Y_k . This matrix can change with time, but here it will be assumed constant.

The random variables W_n equation (4.18), and V_n equation (4.19), represent the process and measurement noise vectors, respectively.

Lets consider that vector X_k contains all the information about the state of the system at time step k , but it cannot be measured directly. Instead, only the Y_k can be measured. It is a function of X_k that is corrupted by the measurement noise V_k , equation (4.19).

The overall objective of the Kalman filter is to estimate $X_{k,k-1}$ by minimizing the error \mathcal{E} , which is a function of the difference between the true state X and on estimated state \hat{X} .

Lets define *a priori* and *a posteriori* estimate error, respectively:

$$\mathcal{E}_{k,k-1} = X_{k,k} - \hat{X}_{k,k-1} \quad (4.20)$$

$$\mathcal{E}_{k,k} = X_{k,k} - \hat{X}_{k,k} \quad (4.21)$$

where $\hat{X}_{k,k-1}$ is *a priori* state estimate vector at time k given knowledge of the process at time $k-1$, and $\hat{X}_{k,k}$ is *a posteriori* state estimate vector at time k given measurement Y_k .

The error \mathcal{E} should be positive and increase monotonically. So the error function should be a squared error function as follows:

$$\mathcal{E}_{k,k-1} = (X_k - \hat{X}_{k,k-1})^2 \quad (4.22)$$

$$\mathcal{E}_{k,k} = (X_k - \hat{X}_{k,k})^2 \quad (4.23)$$

The *a priori* estimate error covariance matrix, $P_{k,k-1}$, is represented by equation (4.24) and the *a posteriori* estimate error covariance matrix, $P_{k,k}$, is represented by equation (4.25).

$$P_{k,k-1} = E[\varepsilon_{k,k-1}\varepsilon_{k,k-1}^T] \quad (4.24)$$

$$P_{k,k} = E[\varepsilon_{k,k}\varepsilon_{k,k}^T] \quad (4.25)$$

where $E(\cdot)$ means the expected value.

For minimization of the mean squared error in order to yield the optimal filter, it is assumed that is possible to correctly model the system errors using Gaussian distributions. Assuming that W_k and V_k are independent of each other, equation (4.26), white and with normal probability distributions (mean equal to zero), equation (4.27):

$$E[V_k W_k] = 0 \quad (4.26)$$

$$p(w) \sim N(0, Q), \quad (4.27)$$

$$p(v) \sim N(0, R).$$

The process noise and measurement noise covariance, Q and R , respectively, are assumed stationary over time k , i.e the distribution of the errors around its mean does not change over time.

It is possible to write an equation for the new estimate, $\hat{X}_{k,k}$, combining the old estimate, $\hat{X}_{k,k-1}$ with the measurement data. So, to compute an *a posteriori* state estimate vector $\hat{X}_{k,k}$ as a linear combination of an *a priori* estimate vector $\hat{X}_{k,k-1}$ and a weighted difference between an actual measurement vector Y_k and a measurement prediction $H\hat{X}_{k,k-1}$ it is used the equation (4.28).

$$\hat{X}_{k,k} = \hat{X}_{k,k-1} + K_k (Y_k - H\hat{X}_{k,k-1}) \quad (4.28)$$

The difference $(Y_k - H\hat{X}_{k,k-1})$ is designated by *measurement innovation* and reflects the discrepancy between the predicted measurement $H\hat{X}_{k,k-1}$ and the actual measurement vector Y_k . An innovation of zero means that the two are in complete agreement.

K_k is $n \times m$ matrix designated by gain that minimizes (4.25). This minimization is performed as a function of the *a priori* covariance error, $P_{k,k-1}$:

$$K_k = P_{k,k-1} H^T (H P_{k,k-1} H^T + R)^{-1} \quad (4.29)$$

K_k defines the credibility of the measurement vector Y_k .

Then, the new *a posteriori* covariance error, $P_{k,k}$, can be found through the following equation:

$$P_{k,k} = (I - K_k H) P_{k,k-1} \quad (4.30)$$

I refers to the identity matrix.

The equations for the Kalman filter are divided into *time update or predictor equations* and *measurement update or corrector equations*.

The time update equations are responsible for projecting forward (in time) the current state, $\hat{X}_{k-1,k-1}$, and error covariance, $P_{k-1,k-1}$, to obtain the *a priori* estimates for the next time k . According to some derivations from equations (4.18) and (4.24), these equations are represented by equations (4.31) and (4.32).

$$\hat{X}_{k,k-1} = A \hat{X}_{k-1,k-1} \quad (4.31)$$

$$P_{k,k-1} = A P_{k-1,k-1} A^T + Q \quad (4.32)$$

$X_{k-1,k-1}$ and $P_{k-1,k-1}$ have to be initialized.

The *measurement update equations* are responsible for incorporating a new measurement into the *a priori* estimate to obtain an improved *a posteriori* estimate, *i.e.* feedback. These equations are represented by equations (4.28), (4.29) and (4.30).

$$\hat{X}_{k,k} = \hat{X}_{k,k-1} + K_k (Y_k - H \hat{X}_{k,k-1}) \quad (4.28)$$

$$K_k = P_{k,k-1} H^T (H P_{k,k-1} H^T + R)^{-1} \quad (4.29)$$

$$P_{k,k} = (I - K_k H) P_{k,k-1} \quad (4.30)$$

After each time and measurement update pair, the process is repeated with the previous *a posteriori* estimates used to predict the new *a priori* estimates. This is represented in figure 4.11. This recursive nature makes practical implementations much more feasible than, for example, Wiener filter [98] which is designed to operate on all of the data directly for each estimate. The Kalman filter instead recursively conditions the current estimate on all of the past measurements.

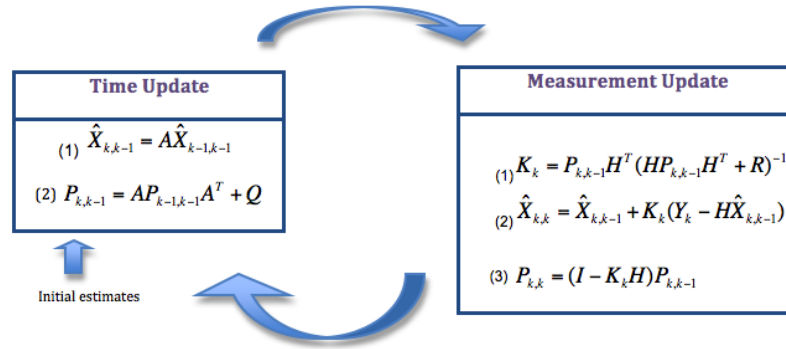


Figure 4.11 A complete Picture of the operation of the Kalman filter.

In the current problem, consider the system dynamic equations [99]:

$$\begin{aligned} x_{k+1} &= x_k + T\dot{x}_k, \\ \dot{x}_{k+1} &= \dot{x}_k + u_k \end{aligned} \quad (4.33)$$

In these equations it is presented a stochastic model that considers a first derivative influenced by a random noise u_k , *i.e.* first derivative is not constant.

The equation that links the actual state x_k and the measured y_k is called the *observation equation*:

$$y_k = x_k + v_k \quad (4.34)$$

Thus the filter initial estimate parameters are defined as following:

$$A = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, H = [0 \quad 1], V_k = [v_k] \text{ and } W_k = \begin{bmatrix} 0 \\ w_k \end{bmatrix},$$

being,

$$\hat{X}_{k,k} = \begin{bmatrix} \hat{x}_{k,k} \\ \hat{\dot{x}}_{k,k} \end{bmatrix}, \hat{X}_{k+1,k} = \begin{bmatrix} \hat{x}_{k+1,k} \\ \hat{\dot{x}}_{k+1,k} \end{bmatrix} \text{ and } Y_k = [y_k].$$

Then, the Kalman filter parameters, R and Q , have to be tuned.

In the actual implementation of the filter, the measurement noise covariance R is measured prior to operation of the filter. Measuring the measurement error covariance R is practical because generally it is simple to take some off-line sample measurements in order to determine the average variance of the measurement noise,

$$R = [\sigma_x^2] \quad .$$

Through equation (4.29) one can see that as the measurement error covariance R decreases, the gain K_k increases and will weigh the innovation $(Y_k - H\hat{X}_{k,k-1})$ more heavily. Thus, it is given a lot of credibility to the actual measurement vector Y_k , while the predicted measurement $H\hat{X}_{k,k-1}$ is trusted less and less. On the other hand, as the *a priori* estimate error covariance $P_{k,k-1}$ approaches to zero, the gain K_k weights the innovation less heavily. Thus, the actual measurement Y_k is trusted less and less, while the predicted measurement $H\hat{X}_{k,k-1}$ is trusted more and more.

The determination of the process noise covariance Q is generally more difficult as it is difficult to have the ability to directly observe the process that is being estimated. Sometimes a relatively simple process model can produce acceptable results if one “injects” enough uncertainty into the process via the selection of Q .

In this application the process noise covariance Q is formulated based on the first derivative noise, which affects the estimation of the user’s command intentions. The value of Q is related to the process error of the system. This represents the fact that real models are different than ideal models. Thus, a good choice of Q helps the filter to estimate more precisely the true state.

It is calculated using off-line measures of the signal. For each measure the covariance of the signal is calculated. Finally, the process noise covariance is the average of all the calculated covariances. A piecewise small constant first derivative model is considered [99]:

$$Q = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_v^2 \end{bmatrix},$$

where σ_v^2 is the variance of the random first derivative component of the process noise.

After tuning these two parameters, the initial covariance error $P_{k-1,k-1}$ is set equal to Q , because before the initiation of the state variables estimation the process error already exists and the initial estimated state $\hat{X}_{k-1,k-1}$ equal the initial state X_0 .

These initial definitions allow obtaining the time update equations (4.31) and (4.32). Then calculating the Kalman gain K_k using equation (4.29) and measuring the output Y_k , it is calculated the updated state estimation (4.28).

Finally the covariance $P_{k,k}$ is calculated through (4.30) and a new iteration starts beginning a new cycle.

To understand better the importance that the choice of Q and R have on the Kalman filter behaviour, two different values for these parameters will be given. The influence that they have in the Kalman gain K_k and in the a priori and a posteriori covariance error, $P_{k,k-1}$ and $P_{k,k}$, will be verified.

For $Q = 2 \times 10^{-9}$ and $R = 1 \times 10^{-5}$, the measured state, the estimated state and the true state are shown in figure 4.12. To clarify, the measured state will be “raw Z” and it is the process containing both measure and process errors (V_k and W_k); the estimated state will be “Z filt Kalman” and it is the output of the Kalman filter $\hat{x}_{k,k}$; the true state will be “ideal Z” and it is the given process but without V_k and W_k .

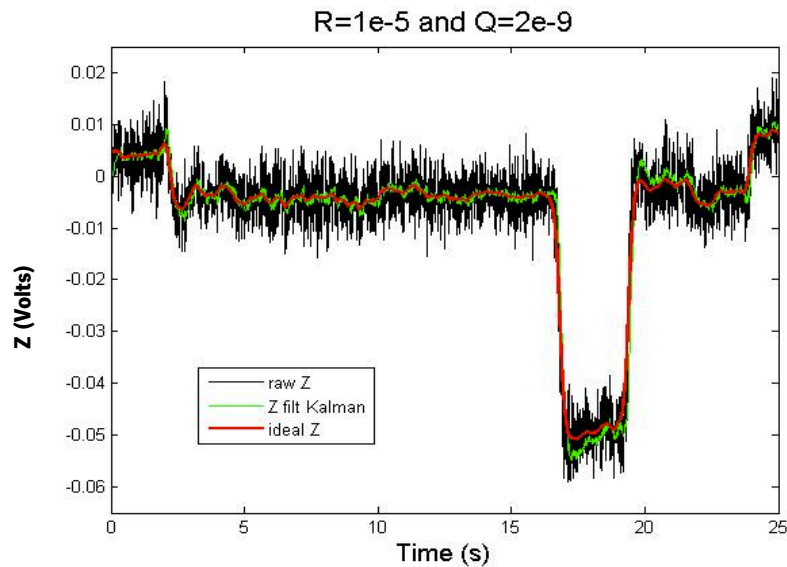


Figure 4.12 Measured state (raw Z) in black; estimated state (Z filt Kalman) in green and true state (ideal Z) in red.

If the Kalman filter had a perfect performance, the estimated state (Z filt kalman) would coincide with the true state (ideal Z). However, the error between the true state and the measured state is higher than the error between the true state and the estimated state, as it can be seen in figure 4.13, which means that the Kalman filter had a good performance as a filter, attenuating the noise components.

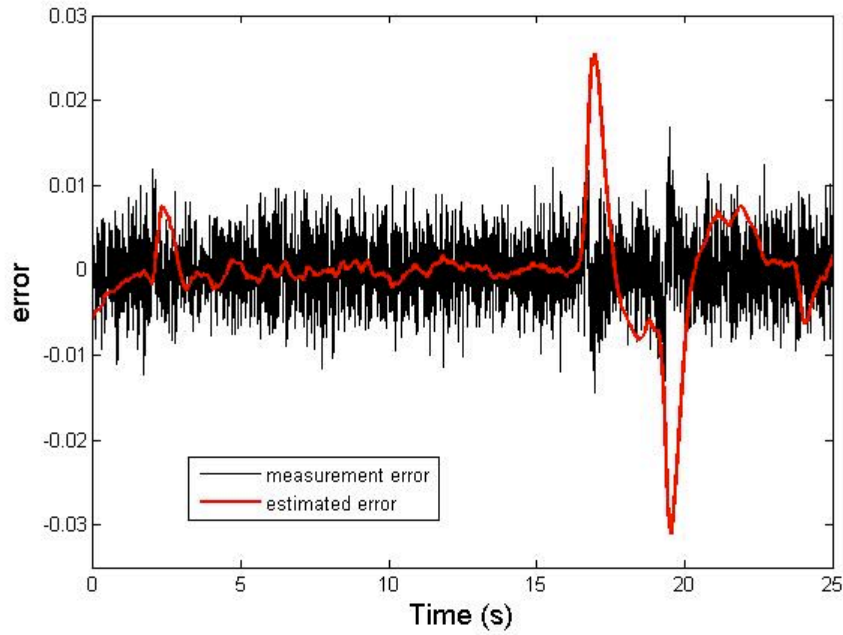


Figure 4.13 Black: error between the true state and the measured state. Red: error between the true state and the estimated state.

Figure 4.14 presents the results for the Kalman gain, K_k , a priori covariance error $P_{k,k-1}$ and a posteriori covariance error, $P_{k,k}$.

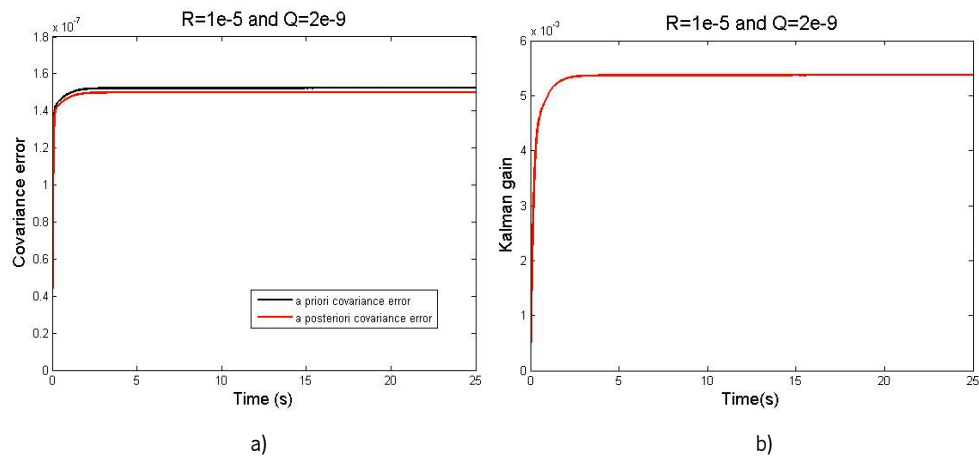


Figure 4.14 a) The *a priori* error covariance (black) and the *a posteriori* error covariance (red); b) The kalman gain.

In figure 4.14a one can see that the a priori covariance error, $P_{k,k-1}$, is higher than the a posteriori covariance error, $P_{k,k}$. This means that the error after the measurement stage (a posteriori covariance error, $P_{k,k}$) was reduced with the help of the Kalman gain, K_k (Figure 4.14b).

This conclusion is based in the fact that K_k is used to minimize the a posteriori error covariance, $P_{k,k}$, until both reach a steady-state.

If R is increased to 1×10^{-3} , it can be verified in Figure 4.15 that the estimated state (Z filt kalman) approaches more closely the true state (ideal Z). This means that the kalman filter believes less in the measurements and more in the estimations. However, this R value introduces a significant delay on the estimate state, meaning that this is not a perfect behavior of the kalman filter.

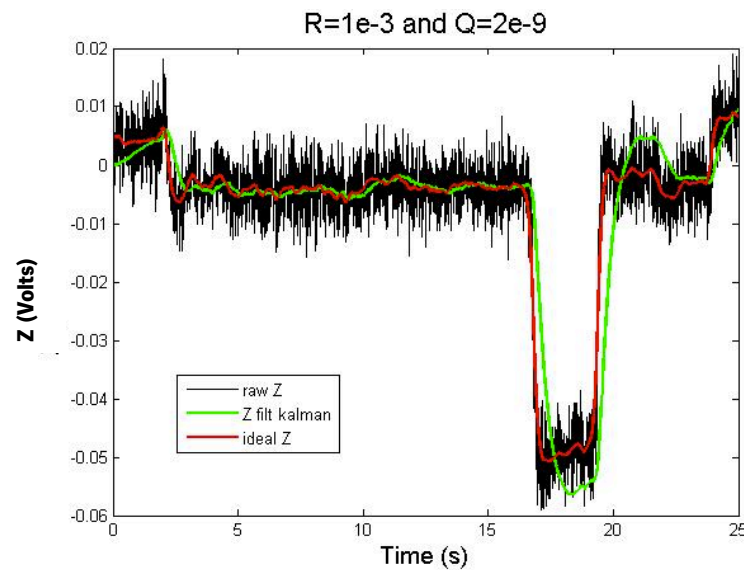


Figure 4.15 Measured state (raw Z) in black; estimated state (Z filt Kalman) in green and true state (ideal Z) in red.

Figure 4.16 shows that the error between the true and the estimated state is closest to zero than the one represented in figure 4.13, which means that the Kalman filter had a better performance as a filter, attenuating the noise components.

Looking to figure 4.17b, and compare it with figure 4.14b, it can be seen that the Kalman gain, K_k , was reduced with the increase of the R value, given less accuracy to the measurements.

It can also be seen that the difference between the a priori covariance error, $P_{k,k-1}$, and the a posteriori covariance error, $P_{k,k}$, is minimum, however the a priori covariance error is still higher than the a posteriori. This can be explain by the reduced Kalman gain and by the fact that the a priori covariance error initial value is equal to Q , considering that the process error already exists in the beginning of the measurement state.

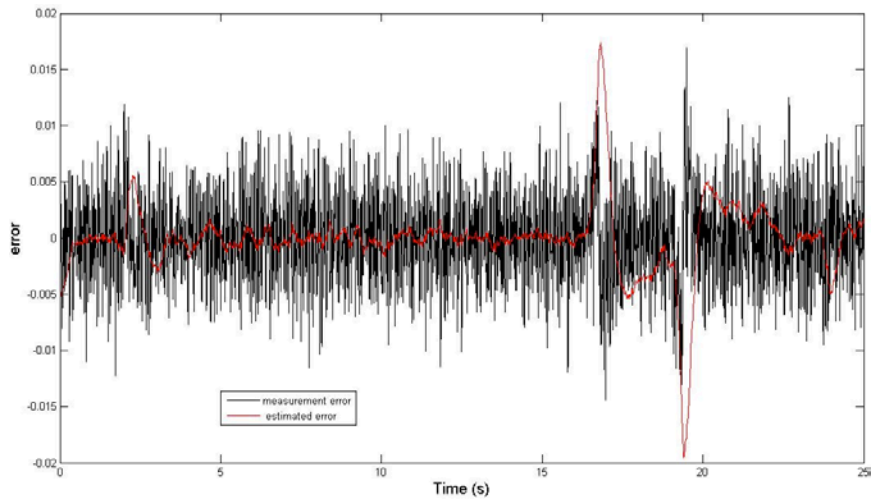


Figure 4.16 Black: error between the true state and the measured state. Red: error between the true state and the estimated state.

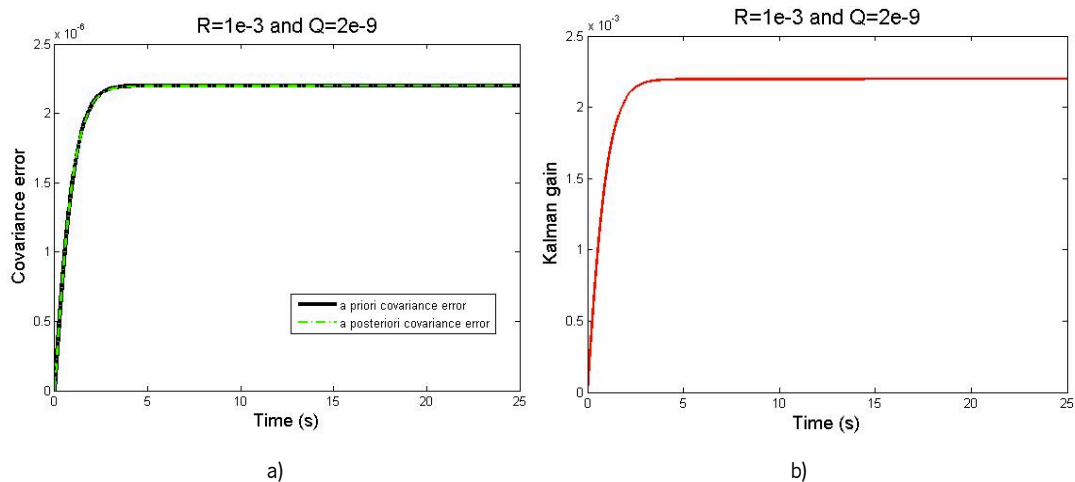


Figure 4.17 a) The *a priori* error covariance (black) and the *a posteriori* error covariance (green); b) The kalman gain.

The convergence of the covariance error increased in comparison with the figure 4.14a with a smaller R . This is because the estimated state (Z filt kalman) takes more time to believe in the measurements (raw Z), following closely the estimated state.

This time of convergence can be fundamental to the quality of the estimation of the Kalman filter. If it takes more time to estimate (higher time of convergence), it could achieve a better estimation. However, this increase of convergence time can introduce a significant delay between the measured state (raw Z) and the filtered one (Z filt kalman), as it can be seen in figure 4.15.

So, it is needed to found a compromise between the quality of estimation and the delay introduced on the filtered signal, when the filter is designed.

If the Q value is increased the result states are presented in figure 4.18.

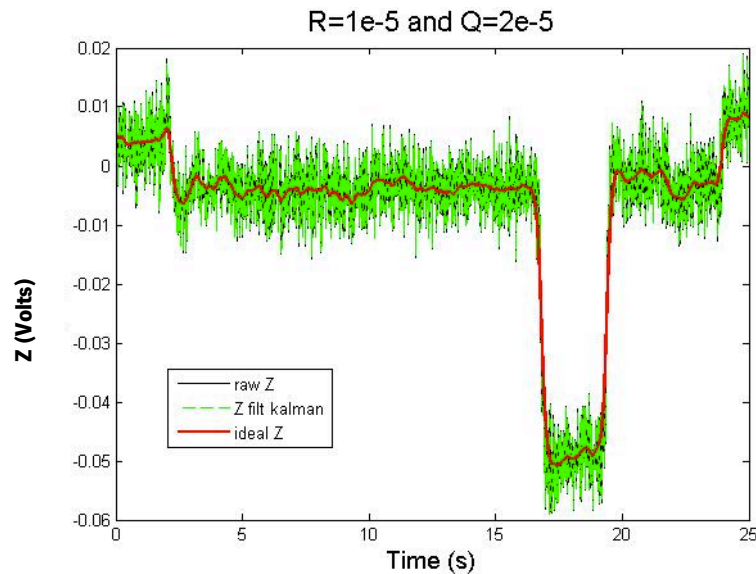


Figure 4.18 Measured state (raw Z) in black; estimated state (Z filt Kalman) in green and true state (ideal Z) in red.

Looking at figure 4.19, one can see that in this case the error between the true state (ideal Z) and the estimated state (Z filt kalman) is maximum, which means that the Kalman filter has a very bad performance as a filter, not attenuating the noise components. Comparing with the other results (Figures 4.13 and 4.16), it can be seen that these is the worst result from the three, presenting the highest error the true state and the estimated state.

A good choice of Q helps the filter to estimate more precisely the true state. Which in this case that does not happen because the value of Q is higher than the optimal one. As figure 4.19 shows, the estimated state (Z filt kalman) is closer to the measured state (raw Z) than the true state (ideal Z), representing a bad performance of the kalman filter.

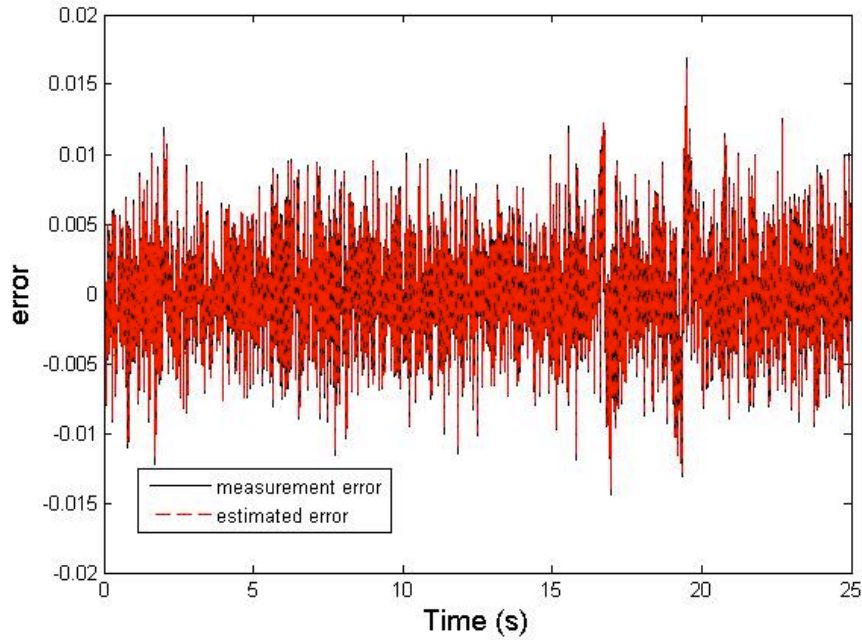


Figure 4.19 Black: error between the true state and the measured state. Red: error between the true state and the estimated state.

It is also noteworthy that the higher Q the higher is the convergence of the Kalman filter, *i.e.* the Kalman gain, K_k and the covariance errors, $P_{k,k-1}$ and $P_{k,k}$ (Figure 4.20).

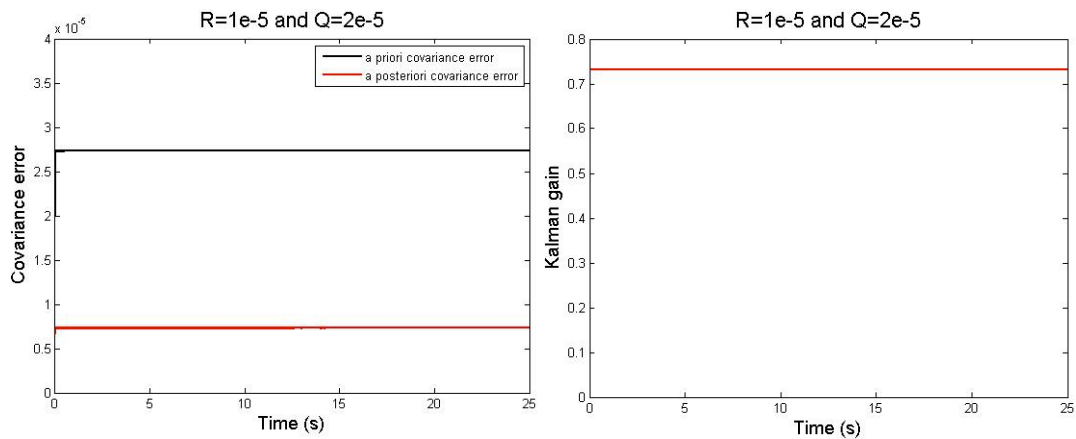


Figure 4.20 a) The *a priori* error covariance (black) and the *a posteriori* error covariance (red); b) The kalman gain.

With this brief study about the behavior of the kalman filter by varying the Q and R values, it can be concluded that the increase of the value of the measurement noise covariance R , decreased the kalman gain, as it was expected by the equation (4.29). Therefore, the measured state will be less trusted than the estimated one making the Kalman filter output to be closely to the true value, than

the measured one. On the other hand, if R is decreased, the kalman gain increase, equation (4.29) and the kalman output will trust more in the measurements.

The R value should be set as real as possible, being the average variance of the noise components. If R is higher than it should be, the kalman filter will discard, perhaps, valuable information about the true state of the system. And if R is lower than the true covariance, the kalman filter will trust in false information that does not correspond to the true state, but to the noise state, using erroneous information about the system.

It can also be conclude that the variation of R and Q influences the convergence of the Kalman filter. A smaller R indicates a faster convergence of the Kalman filter, because the Kalman will believe quickly in the measurements and will converge to a stable error covariance; and a higher R indicates a slower converge of the Kalman filter.

On the other hand, the higher Q the higher is the convergence of the Kalman filter, and the smaller Q the slower is the convergence.

A good choice has to be done, so that the kalman presents a good performance on eliminating/attenuating the noise components from the signal, by taking a trade-off between velocity of estimation and accuracy.

When compared the g-h filter with the Kalman filter one can see that the g-h is a simplification of this filter. The g-h uses a deterministic description for the behaviour of the signal, considering a constant first derivative model, and the Kalman filter considers a stochastic model [99]. However, there are similarities between these filters.

If the kalman gain is set constant and equal to:

$$K = \begin{bmatrix} g \\ h \\ T_s \end{bmatrix},$$

The Kalman filter becomes a g-h filter, if K is assumed constant and not adaptative.

4.4.1.1. Selection of the filters parameters and their optimization

The selection of the BBF parameter g , CDF parameter θ and Kalman filter parameter R and Q , is presented in this section.

For this selection the Kinematic Estimation Error (KTE) was used. KTE evaluates the smoothness, response time, and execution time of a tracking algorithm [105] and is expressed by:

$$KTE = \sqrt{|\bar{\varepsilon}|^2 + \sigma^2} \quad (4.35)$$

$|\bar{\varepsilon}|^2$ and σ^2 are the mean and variance of the absolute estimation error between a desired signal and the measured signal. The desired signal is obtained by filtering offline the signals measurements with a Butterworth filter, *i.e.* zero-phase forward and reverse digital filtering. This filter provides for the reference signal.

To select the filters parameters (g for BBF, θ for CDF, R and Q for Kalman), five individuals drove the walker without any motorization executing three experiments (Table 4.1) with five repetitions each.

During the tests the signals of the joystick were acquired.

These signals were then introduced off-line in the filters algorithm (Kalman filter, BBF and CDF) using a broad range of g , θ , R and Q parameters. The result was processed by the KTE. The best solutions for each filter, *i.e.* the ones with the lowest KTE, were chosen for each user, experiment and repetition. With these results, it was calculated the mean of the best 75 solutions for each parameter, as well as the mean of the delay between the input and the output for each case.

i) g parameter

The g parameter of BBF filter was varied between $0.001 < g < 0.1$, with a step of 0.0001, before experimenting some values to observe when a good attenuation of the noise was achieved.

Table 4.4 presents the g parameters of the BBF filter, the corresponding delay between the Y-signal and the filtered signal and the KTE, organized by subjects (1,2,3,4,5) and A,B,C trajectories. Table 4.5 is similar but to the Z-signal.

In table 4.4, it can be seen that the average g for all the experiments for the Y-signal is 44.29×10^{-3} , with a standard deviation of 4.97×10^{-3} . The average KTE presents the value of 6.14×10^{-3} , which is a small error value, showing a good performance of the filter. The average delay between the estimated signal (filtered signal) and the measured signal is also small, 3.7 ms. This delay is about 60 times inferior to the time of human perception (200 ms), not causing prejudice to the human-machine interaction. This means that BBF filter has high quality for this application.

The dispersion of the g parameters (4.97×10^{-3}) is small compared with its average, meaning that this value does not have a big variation between subjects and trajectories.

Table 4.4 Average values* of the best solutions of the g parameter (BBF), delay between the Y original signal and the filtered signal and KTE for each Y joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.

Subject	Trajectory	g ($\cdot 10^{-3}$)	KTE ($\cdot 10^{-3}$ rad/s)	Delay ($\cdot 10^{-3}$ s)
1	A	30.00 \pm 2.85	4.24 \pm 0.57	5.8 \pm 2.17
	B	31.14 \pm 4.69	4.64 \pm 0.55	6.0 \pm 2.0
	C	39.16 \pm 3.62	7.78 \pm 1.49	5.6 \pm 1.67
2	A	58.10 \pm 4.80	9.33 \pm 0.89	0.0 \pm 0.0
	B	58.94 \pm 5.13	7.42 \pm 0.93	0.2 \pm 0.44
	C	57.26 \pm 5.09	5.70 \pm 0.42	0.2 \pm 0.44
3	A	32.70 \pm 4.07	3.92 \pm 0.03	4.4 \pm 1.14
	B	42.94 \pm 5.38	7.04 \pm 0.97	4.2 \pm 1.48
	C	55.20 \pm 2.98	7.08 \pm 1.89	2.6 \pm 1.52
4	A	59.88 \pm 9.94	5.78 \pm 0.48	0.0 \pm 0.0
	B	42.24 \pm 5.73	8.02 \pm 1.03	4.4 \pm 1.52
	C	63.10 \pm 11.83	6.86 \pm 0.82	1.4 \pm 1.14
5	A	28.14 \pm 2.26	3.98 \pm 0.49	6.0 \pm 2.0
	B	31.15 \pm 3.03	3.70 \pm 0.31	4.6 \pm 1.82
	C	34.46 \pm 3.23	4.60 \pm 0.82	5.8 \pm 1.48
Total Mean		44.29\pm4.97	6.14\pm0.78	3.7\pm1.25

*Each mean corresponds to the mean of the 5 repetitions that each user performed for each trajectory.

In table 4.5, it is observed that the average g for all the experiments for the Z-signal is 16.87×10^{-3} , with a standard deviation of 2.51×10^{-3} . The average KTE presents the value of 1.64×10^{-3} , which is a small error value. Thus, the filter had a good performance. The average delay between the estimated signal (filtered signal) and the measured signal is also small, 24 ms. This delay is about 10 times inferior to the time of human perception (200 ms), which means that the BBF filter showed is also suitable for filtering the Z signals in this application.

The dispersion of the g parameters (2.51×10^{-3}) is small comparing with its average, meaning that its value does not have a big variation between subjects and trajectories.

Table 4.5 Average values* of the best solutions of the g parameter (BBF), delay between the Z original signal and the filtered signal and KTE for each Z joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.

Subject	Trajectory	g ($\cdot 10^{-3}$)	KTE ($\cdot 10^{-3}$ rad/s)	Delay($\cdot 10^{-3}$ s)
1	A	17.18 \pm 6.13	1.56 \pm 0.59	24.6 \pm 4.28
	B	15.70 \pm 2.58	1.60 \pm 0.18	23.2 \pm 6.45
	C	20.88 \pm 3.00	1.62 \pm 0.20	19 \pm 4.06
2	A	17.50 \pm 3.83	1.38 \pm 0.09	25.5 \pm 3.69
	B	15.77 \pm 3.25	1.37 \pm 0.01	24.6 \pm 5.18
	C	19.82 \pm 2.38	1.02 \pm 0.08	43.6 \pm 17.28
3	A	18.86 \pm 1.37	1.64 \pm 0.16	20.0 \pm 1.87
	B	20.42 \pm 2.80	1.84 \pm 0.16	17.8 \pm 3.42
	C	14.10 \pm 1.77	1.38 \pm 0.16	24 \pm 2.55
4	A	12.28 \pm 1.00	3.10 \pm 4.41	42.2 \pm 7.52
	B	19.56 \pm 2.67	1.46 \pm 0.18	20.6 \pm 5.68
	C	16.10 \pm 0.54	1.72 \pm 0.08	22 \pm 1.22
5	A	10.54 \pm 0.63	1.32 \pm 0.04	32.0 \pm 2.12
	B	10.00 \pm 0.74	1.36 \pm 0.08	32.8 \pm 1.30
	C	24.02 \pm 5.15	1.92 \pm 0.27	16.2 \pm 5.11
Total Mean		16.87\pm2.51	1.64\pm0.44	24.0\pm4.78

*Each mean corresponds to the mean of the 5 repetitions that each user performed for each trajectory.

Comparing the two values of average g for each signal (Y and Z), it can be verified that the average g parameter of the Z -signal compared with the average g parameter of the Y -signal shows a lower value. These results were expected, since the SNR (Table 4.2) of the Z -signal is much lower than the Y -signal, requiring to be further filtered. Thus, in the Z -signal case the filter will believe less in the measurements and more in the estimated signal (filtered signal).

iii) θ parameter

The θ parameter of the CDF filter was varied between $0.95 < \theta < 1$, with a step of 0.001, before an extensive analysis of the acquired data to detected when a good noise attenuation was performed by the filter.

Table 4.6 presents the θ parameters of the CDF filter, the corresponding delay between the Y -signal and the filtered signal and the KTE, organized by subjects (1,2,3,4,5) and A,B,C trajectories. Table 4.7 is similar but to the Z -signal.

In table 4.7, it can be seen that the average θ for all the experiments for the Y -signal is 0.974, with a standard deviation of 3.85×10^{-3} . The average KTE presents the value of 5.96×10^{-3} , which is a

small error value, showing a good performance of the filter. The average delay is small, 4 ms, which means that it is 50 times inferior to the time of human perception (200 ms). So the CDF filter has high quality for this application.

The dispersion of the θ parameters (3.85×10^{-3}) is very small compared to its average, meaning that its value does not have a big variation between subjects and trajectories.

Table 4.6 Average values* of the best solutions of the θ parameter (CDF), delay between the Y original signal and the filtered signal and KTE for each Y joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.

Subject	Trajectory	θ	KTE ($\cdot 10^{-3}$ rad/s)	Delay ($\cdot 10^{-3}$ s)
1	A	$0.982 \pm 1.92 \times 10^{-3}$	3.56 ± 1.41	10.4 ± 2.07
	B	$0.981 \pm 2.07 \times 10^{-3}$	4.54 ± 0.49	6.4 ± 1.81
	C	$0.981 \pm 1.30 \times 10^{-3}$	4.42 ± 0.71	7.0 ± 3.00
2	A	$0.951 \pm 4.65 \times 10^{-3}$	8.83 ± 0.41	0.0 ± 0.00
	B	$0.962 \pm 6.89 \times 10^{-3}$	7.45 ± 0.92	0.0 ± 0.00
	C	$0.961 \pm 5.94 \times 10^{-3}$	5.82 ± 0.48	0.0 ± 0.00
3	A	$0.969 \pm 9.93 \times 10^{-3}$	6.98 ± 1.79	2.0 ± 2.35
	B	$0.975 \pm 4.09 \times 10^{-3}$	6.80 ± 0.93	4.8 ± 1.92
	C	$0.978 \pm 2.30 \times 10^{-3}$	7.80 ± 0.49	8.0 ± 2.00
4	A	$0.983 \pm 1.64 \times 10^{-3}$	3.90 ± 0.49	6.4 ± 2.70
	B	$0.982 \pm 1.92 \times 10^{-3}$	3.60 ± 0.32	5.0 ± 2.23
	C	$0.983 \pm 4.65 \times 10^{-3}$	3.84 ± 0.27	5.0 ± 1.41
5	A	$0.975 \pm 4.32 \times 10^{-3}$	7.94 ± 1.09	4.8 ± 1.92
	B	$0.962 \pm 3.44 \times 10^{-3}$	5.76 ± 0.39	0.4 ± 0.54
	C	$0.978 \pm 2.68 \times 10^{-3}$	7.58 ± 1.40	6.4 ± 2.30
	Total Mean	$0.974 \pm 3.85 \times 10^{-3}$	5.96 ± 0.77	4.0 ± 1.62

*Each mean corresponds to the mean of the 5 repetitions that each user performed for each trajectory.

In table 4.7, it can be seen that the average θ for all the experiments for the Z-signal is 0.990, with a standard deviation of 1.10×10^{-3} . The average KTE presents the value of 1.5×10^{-3} , which is a small error value, showing a good performance of the filter. The average delay is small, 25.6 ms, which means that it is 8 times inferior to the time of human perception (200 ms). So the CDF filter has high quality for this application.

The dispersion of the θ parameters (1.10×10^{-3}) is small, meaning that its value does not have a big variation between subjects and trajectories.

Table 4.7 Average values* of the best solutions of the θ parameter (CDF), delay between the Z original signal and the filtered signal and KTE for each Z joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.

Subject	Trajectory	θ	KTE ($\cdot 10^{-3}$ rad/s)	Delay($\cdot 10^{-3}$ s)
1	A	$0.991 \pm 2.05 \times 10^{-3}$	1.44 ± 0.48	26.2 ± 3.83
	B	$0.991 \pm 1.41 \times 10^{-3}$	1.81 ± 0.16	23.8 ± 5.61
	C	$0.991 \pm 0.00 \times 10^{-3}$	1.72 ± 0.01	22.8 ± 1.30
2	A	$0.991 \pm 1.70 \times 10^{-3}$	1.32 ± 0.12	27.8 ± 1.71
	B	$0.992 \pm 1.81 \times 10^{-3}$	1.32 ± 0.01	25.6 ± 4.45
	C	$0.993 \pm 0.84 \times 10^{-3}$	1.04 ± 0.10	40.4 ± 8.35
3	A	$0.989 \pm 0.08 \times 10^{-3}$	1.60 ± 0.18	21.8 ± 1.64
	B	$0.988 \pm 1.34 \times 10^{-3}$	1.76 ± 0.21	18.8 ± 2.86
	C	$0.985 \pm 1.31 \times 10^{-3}$	1.88 ± 0.22	13.0 ± 1.58
4	A	$0.994 \pm 0.04 \times 10^{-3}$	1.28 ± 0.05	30.6 ± 2.51
	B	$0.995 \pm 0.54 \times 10^{-3}$	1.32 ± 0.04	32.6 ± 2.07
	C	$0.992 \pm 1.00 \times 10^{-3}$	1.72 ± 0.08	22.0 ± 1.22
5	A	$0.988 \pm 1.67 \times 10^{-3}$	1.97 ± 1.23	24.6 ± 2.07
	B	$0.994 \pm 1.30 \times 10^{-3}$	0.99 ± 0.08	32.8 ± 1.30
	C	$0.988 \pm 1.52 \times 10^{-3}$	1.52 ± 0.20	20.4 ± 3.21
	Total Mean	$0.990 \pm 1.10 \times 10^{-3}$	1.50 ± 0.21	25.6 ± 2.91

*Each mean corresponds to the mean of the 5 repetitions that each user performed for each trajectory.

Similarly to the BBF filter, the noisier Z-signal requires more filtering. As expected, the θ parameter of the Z-signal is higher than the one of the Y-signal.

iii) R and Q parameters

The R parameter (measurement noise) corresponds to the average variance of the measurement noise. Its average between all data is 8.82×10^{-5} rad².s² for the Y-signal noise and 1.3×10^{-5} rad².s² for the Z-signal noise. Y-signal as a higher R value because the amplitude of its noise in time its higher that the noise of the Z-signal.

Fixing this parameter, it was searched for the best Q parameter to obtain the slowest KTE. By calculating the average variance of the first derivative of the signals it was established a range $1.00 \times 10^{-7} < Q < 9.99 \times 10^{-7}$ for the Y-signal, with a step of 1.00×10^{-8} , and a range $1.00 \times 10^{-9} < Q < 9.99 \times 10^{-9}$ for the Z-signal, with a step of 1.00×10^{-10} , to then evaluate the performance of the Kalman filter.

Table 4.8 and 4.9 presents the mean values of the best solutions of the Q parameter, maintaining R constant, with the respective delay between the original signal (Y and Z , respectively) and the filtered signal (with Kalman) and KTE for each joystick signal.

In table 4.8, it can be seen that the average Q for all the experiments for the Y -signal is 3.21×10^{-7} , with a standard deviation of 5.5×10^{-8} . The average KTE presents the value of 10.84×10^{-3} , which is a small error value, showing a good performance of the filter. The average delay is small, 11.7 ms, which means that it is 17 times inferior to the time of human perception (200 ms). So the Kalman filter has good quality to filter the Y -signal.

The divergence of the Q parameters (5.5×10^{-8}) is smaller than its average, meaning that its value does not have a big variation between subjects and trajectories.

Table 4.8 Average Mean values* of the best solutions of the Q parameter (Kalman), delay between the Y original signal and the filtered signal and KTE for each Y joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.

Subject	Trajectory	Q ($\cdot 10^{-7}$) \pm ($\cdot 10^{-7}$)	KTE ($\cdot 10^{-3}$ rad/s)	Delay($\cdot 10^{-3}$ s)
1	A	2.03 \pm 0.91	6.10 \pm 0.82	12.0 \pm 2.00
	B	2.20 \pm 0.89	6.64 \pm 0.84	13.6 \pm 3.61
	C	1.74 \pm 0.50	6.24 \pm 0.68	14.4 \pm 3.36
2	A	4.98 \pm 0.05	15.33 \pm 4.01	9.0 \pm 0.0
	B	5.34 \pm 0.06	14.29 \pm 1.71	7.4 \pm 0.55
	C	5.98 \pm 0.01	11.26 \pm 3.79	8.7 \pm 6.00
3	A	4.27 \pm 1.25	13.76 \pm 5.46	11.0 \pm 2.82
	B	2.85 \pm 0.58	28.78 \pm 42.71	12.4 \pm 2.07
	C	1.46 \pm 0.25	10.80 \pm 0.92	19.2 \pm 1.92
4	A	3.08 \pm 1.11	4.76 \pm 1.16	12.2 \pm 3.56
	B	1.99 \pm 0.90	5.26 \pm 0.73	11.6 \pm 2.51
	C	2.25 \pm 0.39	5.78 \pm 0.35	11.8 \pm 2.58
5	A	2.28 \pm 0.63	11.66 \pm 1.71	15.0 \pm 2.73
	B	5.98 \pm 0.00	11.30 \pm 1.09	7.6 \pm 0.55
	C	1.64 \pm 0.67	9.96 \pm 1.56	16.8 \pm 2.77
	Total Mean	3.21\pm0.55	10.84\pm4.73	11.7\pm2.47

*Each mean corresponds to the mean of the 5 repetitions that each user performed for each trajectory.

In table 4.9, it can be seen that the average Q for all the experiments for the Z -signal is 3.26×10^{-9} , with a standard deviation of 8.78×10^{-9} . The average KTE presents the value of 1.63×10^{-3} , which is a small error value, showing a good performance of the filter. The average delay is small, 34.8 ms, which means that it is 6 times inferior to the time of human perception (200 ms). So the Kalman filter has good quality to filter the Z -signal.

The divergence of the Q parameters (8.78×10^{-9}) is small, meaning that its value does not have a big variation between subjects and trajectories.

Table 4.9 Average values* of the best solutions of the Q parameter (Kalman), delay between the Z original signal and the filtered signal and KTE for each Z joystick signal organized by subjects (1,2,3,4, 5) and trajectories (A, B, C). Table provides mean \pm standard deviation.

Subject	Trajectory	Q ($\cdot 10^{-9}$) \pm ($\cdot 10^{-9}$)	KTE ($\cdot 10^{-3}$ rad/s)	Delay($\cdot 10^{-3}$ s)
1	A	2.22 \pm 0.74	1.48 \pm 0.42	22.4 \pm 14.7
	B	3.58 \pm 1.39	1.84 \pm 0.24	35.2 \pm 5.22
	C	3.45 \pm 0.54	2.00 \pm 0.12	38.2 \pm 1.64
2	A	2.67 \pm 0.82	1.40 \pm 0.14	37.0 \pm 14.2
	B	2.47 \pm 0.53	1.40 \pm 0.08	29.2 \pm 14.32
	C	1.92 \pm 0.04	1.05 \pm 0.12	41.2 \pm 9.39
3	A	3.93 \pm 0.77	1.84 \pm 0.26	40.0 \pm 3.82
	B	4.91 \pm 1.43	2.16 \pm 0.32	35.0 \pm 3.08
	C	6.90 \pm 0.13	2.38 \pm 0.30	38.2 \pm 1.09
4	A	1.75 \pm 0.05	1.32 \pm 0.04	12.6 \pm 2.40
	B	1.04 \pm 0.06	1.30 \pm 0.07	45.4 \pm 7.02
	C	2.48 \pm 0.80	1.52 \pm 0.24	44.2 \pm 8.19
5	A	5.07 \pm 1.55	1.68 \pm 0.29	34.8 \pm 4.08
	B	1.20 \pm 0.43	0.87 \pm 0.14	43.0 \pm 21.84
	C	5.22 \pm 1.78	1.82 \pm 0.34	34.0 \pm 3.81
Total Mean		3.26\pm8.78	1.63\pm0.20	34.8\pm7.65

*Each mean corresponds to the mean of the 5 repetitions that each user performed for each trajectory.

Table 4.10 resumes all the selected filter parameters and the respective KTE and delay. As it can be seen in Table 4.10, the KTE is very low for all the filters, being the lowest one, the CDF's KTE value, as well as its dispersion. The BBF detains the lowest signals' delay.

In terms of dispersion of parameters values, Kalman has the lowest dispersion for Y-signal and CDF has the lowest dispersion for Z signal.

Since CDF presents the lowest KTE for both signals and a lower dispersion of θ value, it seems to be the best option to choose for this application.

Table 4.10 Best parameter solutions for each filter (BBF, CDF and Kalman).

	Signal	g (BBF)	θ (CDF)	Q (Kalman)
Y	Value	$44.20 \pm 4.97 \cdot (10^{-3})$	$0.974 \pm 3.85 \times 10^{-3}$	$3.21 \pm 0.55 \cdot (10^{-7})$
	KTE ($\cdot 10^{-3}$ rad/s)	6.14 ± 0.78	5.96 ± 0.77	10.84 ± 4.73
	Delay ($\cdot 10^{-3}$ s)	3.7 ± 1.25	4.0 ± 1.62	11.7 ± 2.47
Z	Value	$16.87 \pm 2.51 \cdot (10^{-3})$	$0.990 \pm 1.10 \times 10^{-3}$	$3.26 \times 10^{-9} \pm 8.78 \cdot (10^{-9})$
	KTE ($\cdot 10^{-3}$ rad/s)	1.64 ± 0.22	1.50 ± 0.21	1.63 ± 0.20
	Delay ($\cdot 10^{-3}$ s)	24.0 ± 4.78	25.6 ± 2.91	34.8 ± 7.65

However, these conclusions do not take into consideration the sensibility of the three filters to the variation of the selected average parameters. In order to do so, for the selection of the most suitable filter for this application, it will be ran all data again with the three filters but with the selected average parameters of BBF, CDF and Kalman. The best filter will be the one with the lowest KTE and delay.

Due to the resultant delay, the initial value of $P_{k-1,k-1}$ was set to the average P-steady-state value.

As it was observed in figures 4.14a and 4.17a, there is a time of adaptation for the $P_{k+1,k}$ until it achieves the steady-state. If $P_{k+1,k}$ begins with this value, the time of convergence is diminished, taking less time to estimate, and as a consequence decreases the delay between the measured signal and the filtered signal.

Thus, $P_{k+1,k}$ initial value was substituted by the average of $P_{k+1,k}$ steady-state between the acquired data. For the Y-signal, $P_{k+1,k}$ initial value is set to 0.5245 and for the Z-signal is set to 0.2076. Comparing these values, the covariance error steady-state value of the Z-signal is lower than the Y-signal. As the Z-signal has a lower SNR (Table 4.2), the kalman filter considers trustworthier the estimations than the measures. So, the kalman gain will be lower, decreasing the covariance error value.

Looking at tables 4.11 and 4.12, one can see that the delay decreased using the steady-state value. However, as it was expected (see Kalman filter section), the KTE increased. This happens because the time of convergence decreased, *i.e.* adaptation and estimation, decreasing the performance quality of the filter to attenuate the noise.

Table 4.11 – Comparison between P initial value equal to the steady value, 0.5245, and P initial value equal to Q, using the average values of KTE and Delay, related to Y-signal, for each subject.

Subject		P=0.5245	P=Q=3.21x10 ⁻⁷
1	KTE (.10 ⁻³ rad/s)	6.37±0.31	6.37±0.31
	Delay(.10 ⁻³ s)	9.73±0.90	9.87±0.95
2	KTE (.10 ⁻³ rad/s)	14.50±2.75	14.49±2.77
	Delay(.10 ⁻³ s)	10.48±0.83	10.68±0.63
3	KTE (.10 ⁻³ rad/s)	10.56±0.45	10.53±0.45
	Delay(.10 ⁻³ s)	43.86±6.08	45.1±6.08
4	KTE (.10 ⁻³ rad/s)	5.63±0.27	5.62±0.26
	Delay(.10 ⁻³ s)	8.8±0.92	9.0±0.72
5	KTE (.10 ⁻³ rad/s)	11.44±0.56	11.32±0.51
	Delay(.10 ⁻³ s)	10.2±0.72	11.3±0.92
Total Mean KTE		9.71±0.87	9.66±0.86
Total Mean Delay		16.6±1.89	17.2±1.86

Table 4.12 Comparison between P initial value equal to the steady value, 0.5245, and P initial value equal to Q, using the average values of KTE and Delay, related to Y-signal, for each subject.

Subject		P=0.2076	P=Q=3.26x10 ⁻⁹
1	KTE (.10 ⁻³ rad/s)	1.81±0.23	2.01±0.22
	Delay(.10 ⁻³ s)	37.8±2.80	39.4±0.51
2	KTE (.10 ⁻³ rad/s)	1.34±0.22	1.32±0.19
	Delay(.10 ⁻³ s)	32.9±4.61	32.6±2.85
3	KTE (.10 ⁻³ rad/s)	6.76±0.40	6.72±0.41
	Delay(.10 ⁻³ s)	43.9±2.89	45.1±2.27
4	KTE (.10 ⁻³ rad/s)	4.56±0.06	4.48±0.05
	Delay(.10 ⁻³ s)	29.2±5.12	30.7±4.47
5	KTE (.10 ⁻³ rad/s)	1.55±0.43	1.54±0.44
	Delay(.10 ⁻³ s)	34.3±11.66	36.1±9.94
Total Mean KTE		3.20±0.27	3.12±0.26
Total Mean Delay		35.6±5.42	36.22±4.00

It will be used the results from the column P=Q for the comparison between the three tracking filters.

The results obtained for the KTE and Delay, for each signal, using the selected best solutions for each filter parameter, are shown in tables 4.13 and 4.14, respectively.

Despite the previous conclusion the CDF would be the best option between the filters in study, in tables 4.13 and 4.14 one can see that BBF presents better results for the obtained experimental data. In both signals, it presents the lowest KTE and Delay.

Table 4.13 Comparison between the proposed filters using the average values of the selected parameters for the 5 subjects for the Y signal. The presented average values correspond to the mean between the means of each trajectory .Table provides mean \pm standard deviation.

Subject		$g=44.29 \times 10^{-3}$	$\Theta=0.974$	$R=8.82 \times 10^{-9}$ $Q=3.21 \times 10^{-7}$
1	KTE ($\cdot 10^{-3}$ rad/s)	5.46 \pm 0.82	5.65 \pm 0.25	6.37 \pm 0.31
	Delay($\cdot 10^{-3}$ s)	0.0 \pm 0.0	0.0 \pm 0.0	9.87 \pm 0.95
2	KTE ($\cdot 10^{-3}$ rad/s)	7.64 \pm 1.73	7.89 \pm 1.64	14.49 \pm 2.77
	Delay($\cdot 10^{-3}$ s)	0.0 \pm 0.0	0.0 \pm 0.0	10.68 \pm 0.63
3	KTE ($\cdot 10^{-3}$ rad/s)	7.50 \pm 0.54	7.25 \pm 0.65	10.53 \pm 0.45
	Delay($\cdot 10^{-3}$ s)	0.0 \pm 0.0	1.7 \pm 2.08	45.1 \pm 6.08
4	KTE ($\cdot 10^{-3}$ rad/s)	4.17 \pm 0.11	5.96 \pm 0.17	5.62 \pm 0.26
	Delay($\cdot 10^{-3}$ s)	1.8 \pm 0.56	2.8 \pm 0.34	9.0 \pm 0.72
5	KTE ($\cdot 10^{-3}$ rad/s)	7.36 \pm 1.35	7.30 \pm 1.06	11.32 \pm 0.51
	Delay($\cdot 10^{-3}$ s)	0.9 \pm 0.71	3.3 \pm 2.38	11.3 \pm 0.92
Total Mean KTE		6.46\pm0.91	6.81\pm0.75	9.66\pm0.86
Total Mean Delay		0.5\pm0.25	1.7\pm0.96	17.2\pm1.86

Table 4.14 Comparison between the proposed filters using the average values of the selected parameters for the 5 subjects for the Z signal. The presented average values correspond to the mean between the means of each trajectory .Table provides mean \pm standard deviation.

Subject		$g=16.87 \times 10^{-3}$	$\Theta=0.990$	$R=1.3 \times 10^{-5}$ $Q=3.26 \times 10^{-9}$
1	KTE ($\cdot 10^{-3}$ rad/s)	1.72 \pm 0.03	1.67 \pm 0.14	2.01 \pm 0.22
	Delay($\cdot 10^{-3}$ s)	23.0 \pm 2.13	24.0 \pm 1.40	39.4 \pm 0.51
2	KTE ($\cdot 10^{-3}$ rad/s)	1.40 \pm 0.20	1.64 \pm 0.05	1.32 \pm 0.19
	Delay($\cdot 10^{-3}$ s)	26.1 \pm 3.11	25.9 \pm 1.47	32.6 \pm 2.85
3	KTE ($\cdot 10^{-3}$ rad/s)	5.68 \pm 0.24	5.79 \pm 0.27	6.72 \pm 0.41
	Delay($\cdot 10^{-3}$ s)	21.6 \pm 1.01	24.8 \pm 1.27	45.1 \pm 2.27
4	KTE ($\cdot 10^{-3}$ rad/s)	4.44 \pm 0.06	4.46 \pm 0.08	4.48 \pm 0.05
	Delay($\cdot 10^{-3}$ s)	24.7 \pm 2.75	26.0 \pm 2.11	30.7 \pm 4.47
5	KTE ($\cdot 10^{-3}$ rad/s)	1.42 \pm 0.26	1.39 \pm 0.02	1.54 \pm 0.44
	Delay($\cdot 10^{-3}$ s)	23.7 \pm 3.43	25.1 \pm 1.74	36.1 \pm 9.94
Total Mean KTE		2.93\pm0.16	2.99\pm0.11	3.12\pm0.26
Total Mean Delay		23.8\pm2.90	25.2\pm1.59	36.2\pm4.00

This can also be seen in an example of joystick signal in figures 4.21 and 4.22, where is presented the differences between BBF and CDF and BBF and Kalman. The BBF shows a higher attenuation on the oscillations than the other two filters.

Therefore, the BBF filter is selected as the best option to eliminate the higher-frequency components of the joystick's signals with $g=44.29 \times 10^{-3}$ for the Y-signal and $g=16.87 \times 10^{-3}$ for the Z-signal.

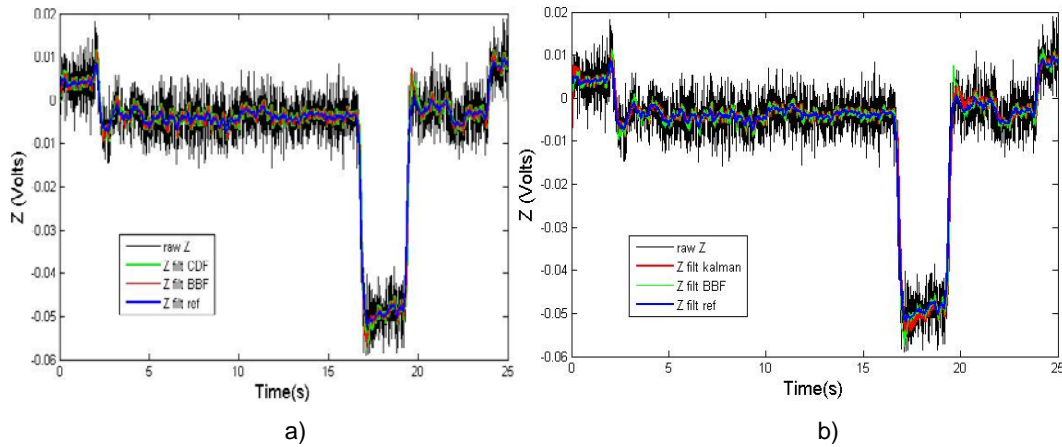


Figure 4.21 a) The superposition of the raw Z with the results of BBF, CDF and Butterworth; b) The superposition of the raw Z with the results of BBF, Kalman and Butterworth.

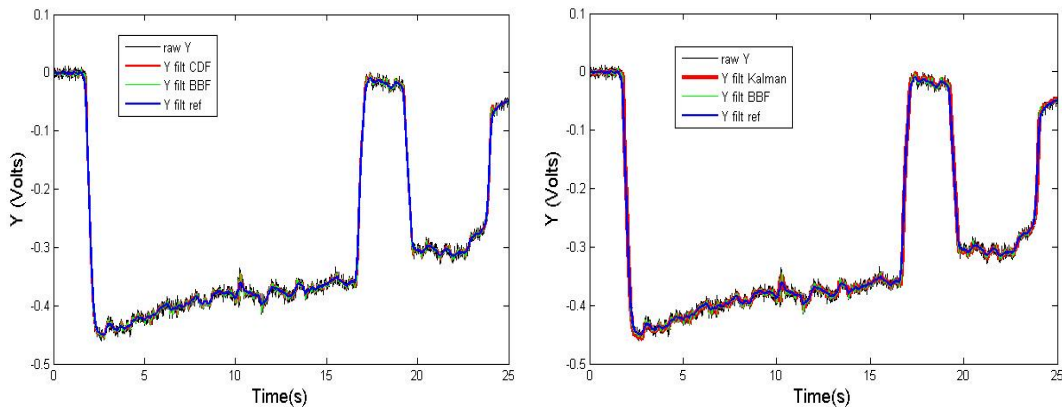


Figure 4.22 a) The superposition of the raw Z with the results of BBF, CDF and Butterworth; b) The superposition of the raw Y with the results of BBF, Kalman and Butterworth.

In figure 4.23 one can observe that the noise was reduced in both signals in comparison with figure 4.7. It also denotes that there are still some oscillations (high-frequency components) in both signals (Figure 4.24), if it is compared with figure 4.8. These oscillations can be reduced by reducing the filter g parameter. However, this change implies an increase in signal delay relative to the reference signal, and therefore worst value of KTE.

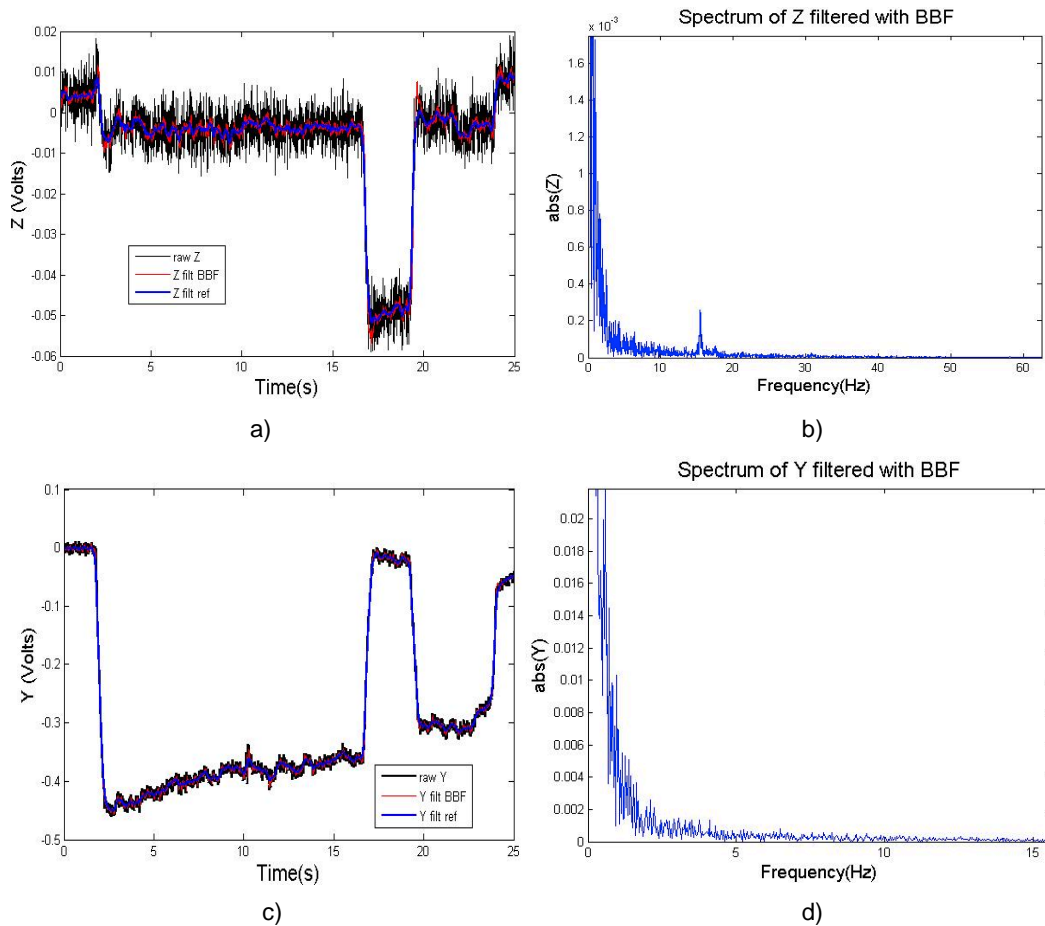


Figure 4.23 Attenuation of higher frequency components by the BBF filter a) The raw Z-signal, the ideal Z and the filtered Z-signal with the BBF filter; b) and d) Frequency Spectrum of Z and Y filtered with BBF, respectively;; c) The raw Y-signal, the ideal Y and the filtered Z-signal with the BBF filter.

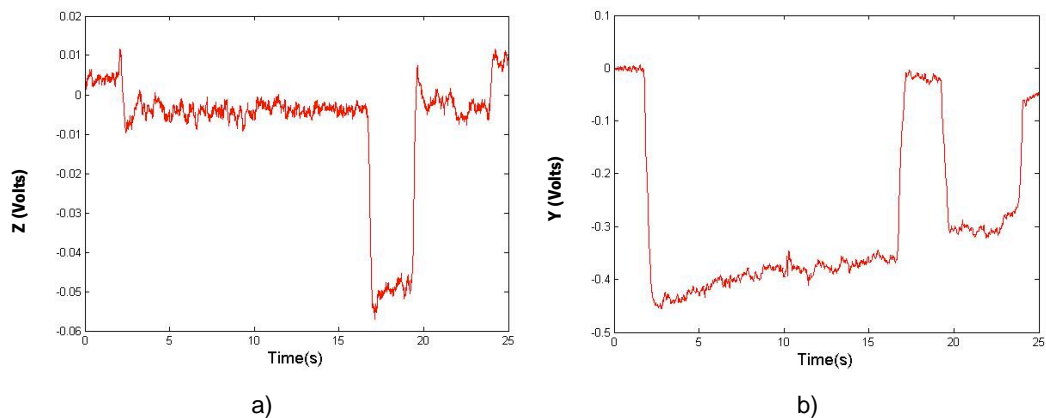


Figure 4.24 a) Z-signal filtered with BBF filter ($g=16.87 \times 10^{-3}$) and b) Y-signal filtered with BBF filter ($g=44.29 \times 10^{-3}$).

In table 4.15, it is also seen that the SNR increased in both signals, meaning that they were filtered, *i.e.* the noise was attenuated.

Table 4.15 SNR of the filtered Y and Z signals.

Signal	SNR
Y	50.05
Z	7.77

4.4.1.1.1. Conclusion of the Filter Strategy

In this section, it was presented a method of user-walker interaction isolate the intentional commands by means of filtering. The interface is based on a joystick and is intended to be user-friendly, simple, efficient, low cost and with little electronics. A series of experiments were performed which showed the sensibility of the joystick to extract navigation commands from the user. These signals presented a higher frequency component that was attenuated by a Benedict-Bordner $g-h$ filter. The g parameter was chosen to be 44.29×10^{-3} for the Y-signal and 16.87×10^{-3} for the Z-signal.

The chosen filter has a low computational cost algorithm, making it a good option to this application, since it can run in a low cost hardware with enough robustness for a commercial device.

4.5. Signal processing architecture of the ASBGo walker

Before the acquisition of the Y and Z signals a pre-processing step was made with hardware. For each signal it was constructed a circuit with three main blocks: Offset block, Filter Strategy block and Amplification block (Figure 4.6).

Each signal has a specific hardware to process it, and it will present in the next sections. After the pre-processing, the signal is acquired by the Arduino platform. These signals were analysed to see if it is necessary to process them with a digital filter strategy.

4.5.1. Y-signal Circuit Hardware

The circuit architecture for the processing of Y-signal is presented in the figure 4.25. In appendix A the full circuit hardware is presented.



Figure 4.25 Diagram of the circuit hardware for the processing of the Y-signal.

The Amplification + Offset block (1) is compound by an inverting sum amplifier using one operation amplifier (electronic specifications are presented in appendix A). This component aims to eliminate the offset from the Y-signal, which value is around 2.5V and then amplify the result signal.

The two inputs correspond to the raw Y signal acquired with the joystick and DC voltage of -2.5 V, which is the offset that Y-signal brings from the joystick that will be summed to the signal eliminating its offset.

It was chose to amplify the signal with a gain of -2.

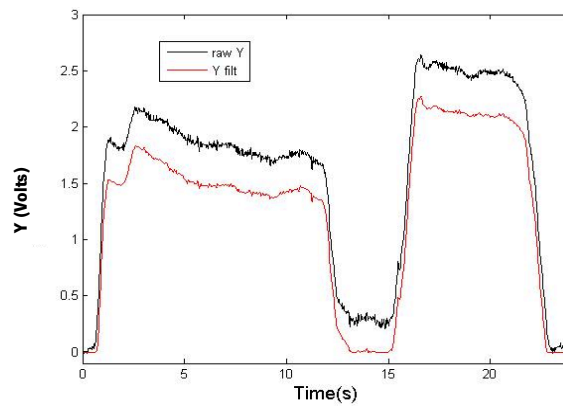
Thus, the result signal will be 2 times amplified and inverted. This inversion is desirable because as the Y-signal has a negative value, this circuit will invert it to positive, making it readable to the Arduino, since it doesn't read negative voltages.

The low pass filter block (2) is to attenuate the higher-frequencies. In this case, as the sampling time is 0.01s, being times smaller than the sampling period of the NeoASAS, the user's command intentions are now located in the range of [0-10Hz]. The electronic specifications are presented in appendix A.

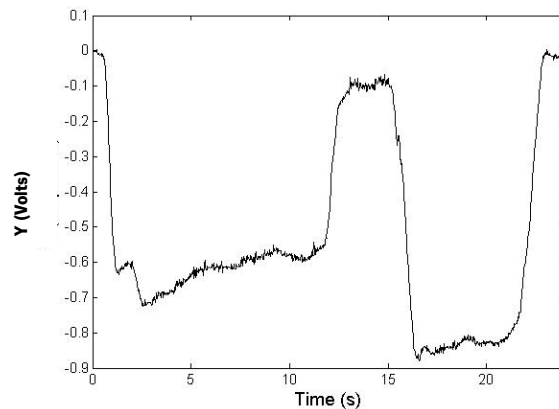
The Amplification block (3) is a non-inverting amplifier (the electronic specifications are presented in appendix A). This block aims to amplify the signal with a gain of 1.5. Thus, the total gain of the circuit is 3, amplifying the signal to a range of [0,3.3], which corresponds to the range that the Arduino platform can read.

A typical output from this circuit is presented in figure 4.26a. This signal can be compared with the one showed in figure 4.26b.

The raw Y in figure 4.26a is a result from the inversion and amplification in Matlab of the Y-signal in figure 4.26b, so that it could be compared with the processed Y-signal represented in figure 4.26a in red (Y filt). With this comparison it can be seen that the processing hardware inverted the signal, added 1.6V (offset) and amplified it successfully. However is denoted a slight attenuation in the Y filt, which means that experimentally the circuit has a gain lower than 3.



a)



b)

Figure 4.26 a) The raw Y-signal processed by software (black) and the Y-signal processed by hardware; b) The raw Y-signal without processing.

In figure 4.27, one can conclude that the signal was successfully filtered, since the high-frequency components were attenuated.

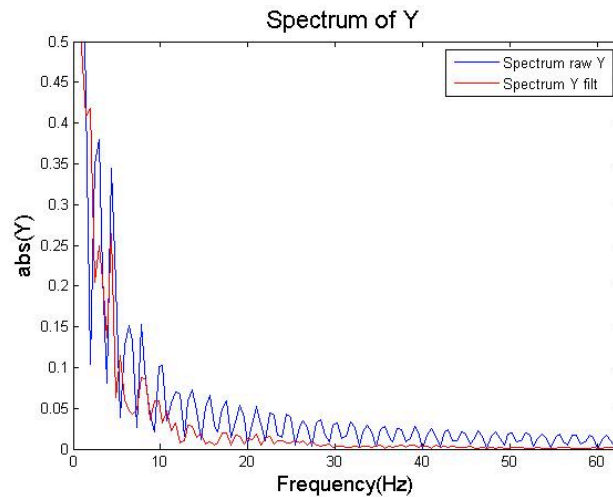


Figure 4.27 Comparison between the Spectrum of the raw Y signal (blue)and the Y-signal filtered by hardware (red)

4.5.2. Z-signal Circuit Hardware

The circuit block for the acquisition of this signal is presented in figure 4.28. In appendix A the completed circuit hardware is presented.

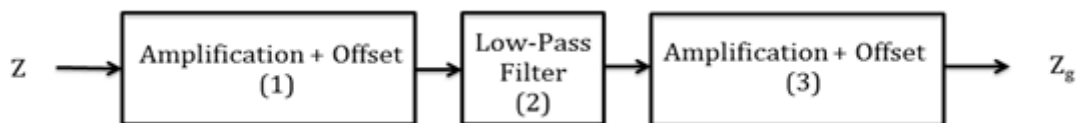


Figure 4.28 Diagram of the circuit hardware for the processing of the Z-signal.

The Amplification + Offset block (1) is also an inverting sum amplifier using one operation amplifier (The electronic specifications are presented in appendix A). The aim is the same as previously, which is to eliminate the offset from the Z-signal, which value is around 2.5V and then amplify the result signal.

The two inputs correspond to the raw Z-signal acquired with the joystick and a DC voltage of -2.5 V, which corresponds to the offset of the Z-signal. A gain of 9 was used to amplify the Z-signal.

The low pass filter block (2) has the same main as the one used in the Y-signal.

The Amplification + Offset block (3) is compound by two amplifiers. First a non-inverting amplifier with a gain equal to 2. Then, an inverting sum amplifier with a gain of 1.

In this case, instead of eliminate the offset from the signal it is summed one equal to 1.6V.

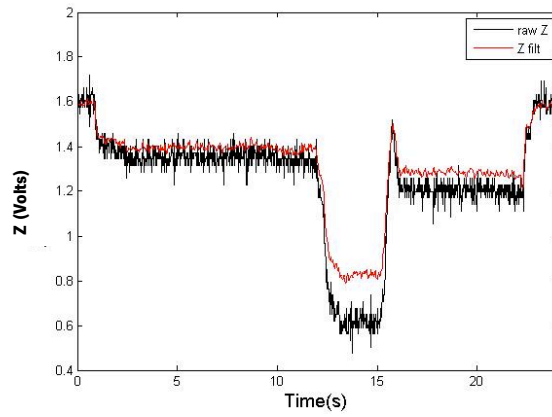
The Z-signal provides the distinction between when the user wants to turn right or left. This distinction is made between a negative or positive sign of the signal. Since the Arduino platform doesn't read negative signals, it is needed to sum an offset to the signal. This offset of 1.6 V is half the maximum voltage that the Arduino can read, which is 3.3V.

The total gain of the circuit is 9, putting the signal into a range of [0,3.3].

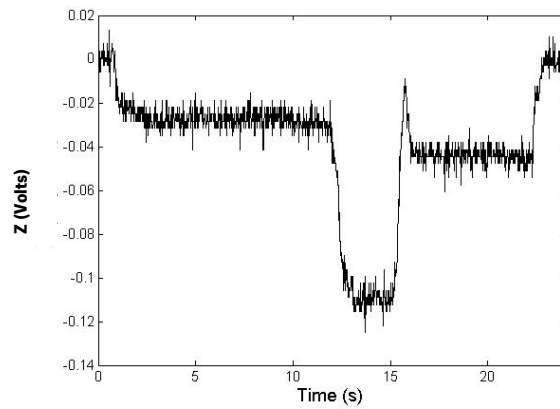
A typical output from this circuit is presented in figure 4.29a. This signal can be compared with the one showed in figure 4.29 b.

The raw Z in figure 4.29a is a result from the inversion and amplification in Matlab of the Z-signal in figure 4.29b, so that it could be compared with the processed Z-signal represented in figure 4.29a in red (Z filt). With this comparison it can be seen that the processing hardware inverted the signal, added 1.6V (offset) and amplified it successfully. However is denoted a slight attenuation in the Z filt, which means that experimentally the circuit has a gain lower than 9.

In figure 4.30, one can conclude that the signal was successfully filtered, since the high-frequency components were attenuated. In this case, the user's command intentions are also located in the range of [0-10Hz].



a)



b)

Figure 4.29 a) The raw Z-signal processed by software (black) and the Z-signal processed by hardware; b) The raw Z-signal without processing.

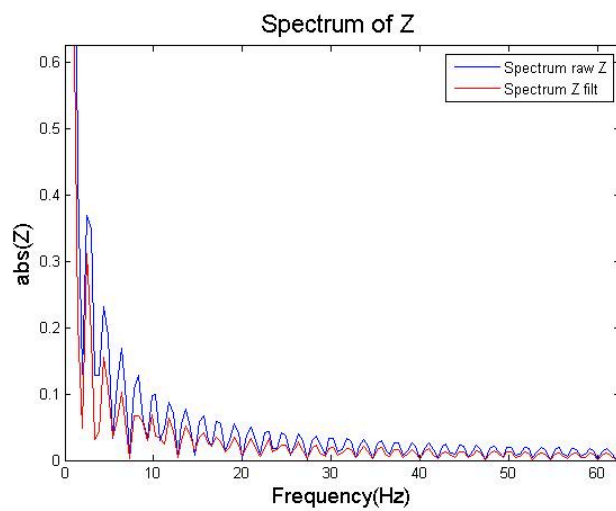


Figure 4.30 Comparison between the Spectrum of the raw Z-signal (blue)and the Y-signal filtered by hardware (red).

4.5.3. Digital Filter Strategy

A hardware circuit was built, and a pre-processing of the joystick's signals was done. As it was seen in the previous section, the signal was amplified and filtered. Comparing the result obtained with the hardware circuit, and represented in figure 4.27a and figure 4.30a, with the ideal signal to be achieved (Figures 4.31a and 4.31c), one can conclude that a digital filter strategy in the Arduino platform is not necessary, since the signals are very similar. Looking also to figures 4.31b and 4.31d, it is assured that in the frequency domain that both signals don't need to be filtered digitally. This will reduce the computational cost of the algorithm that will be implemented in the Arduino platform.

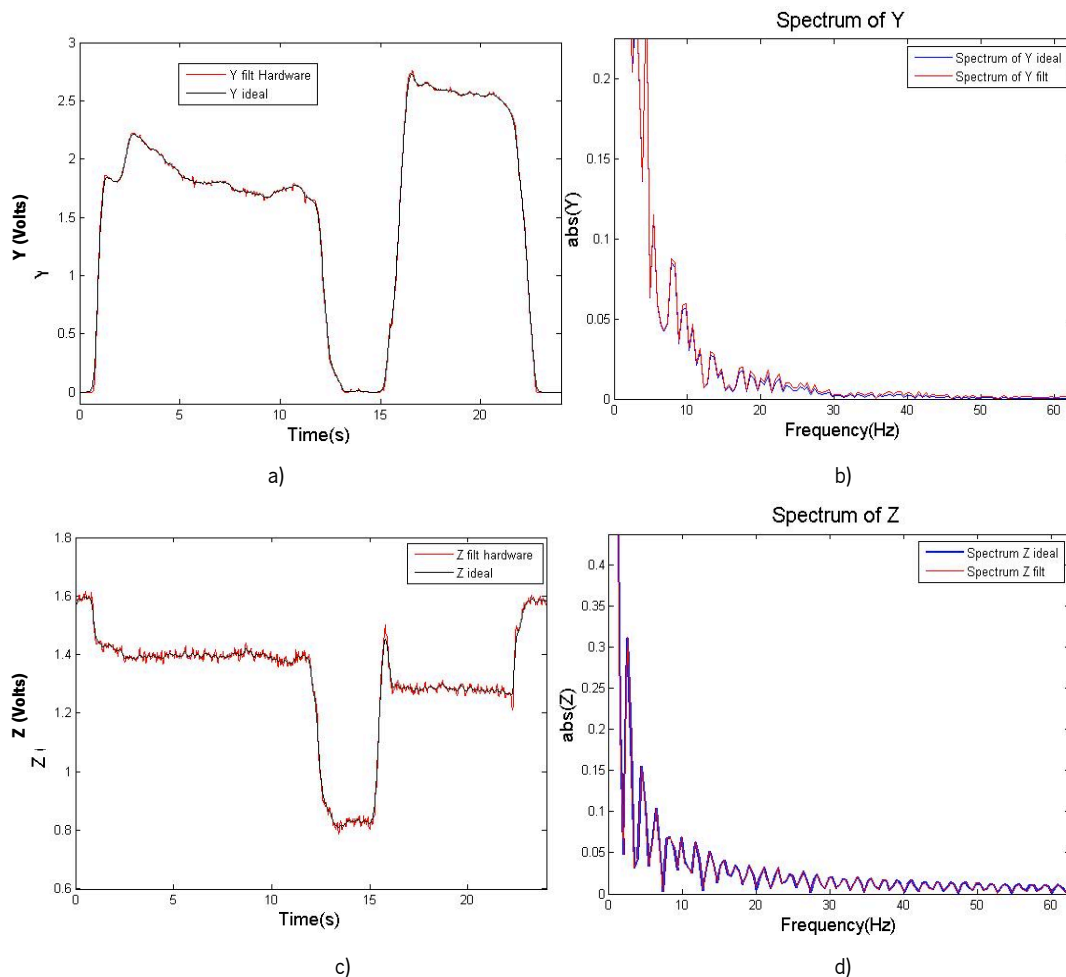


Figure 4.31 Attenuation of higher frequency components by means of recursive digital filters (Butterworth, first order). a) The Y-signal filtered by hardware (YH) in red and the YH filtered with the digital filter in black; b) Frequency Spectrum of Y and Z , respectively;; c) The Z-signal filtered by hardware (ZH) in red and the ZH filtered with the digital filter in black;

4.5.4. Digital Processing of the Joystick's signals.

The Digital processing consist in first acquiring the signal with a sampling period of 0.01s and then transform the range of values of the Y and Z-signals from [0,3.3] to the range [0,1] for the Y-signal and [-1,1] for the Z-signal to then be used on a control strategy to control the motors of the walker, as it is going to be discussed in chapter 6.

In order to accomplish this, the offset of 1.6V is eliminated from the Z-signal (it is not needed to eliminate the offset from the Y-signal since it is 0V) and then divide the Y-signal by 3.3 and the Z-signal by 1.6.

4.6. Conclusion of the Acquisition and Processing Architecture of the NeoASAS and ASBGo walker.

The developed architecture for each walker was successfully implemented and tested. This is necessary to move forward in the workflow of the development of the walkers, since the acquisition and processing of the joystick's signals are fundamental to develop the control strategy that will control the motors' movement.

The overall architecture implemented in NeoASAS and ASBGo is shown in figures 4.32 and 4.33, respectively.

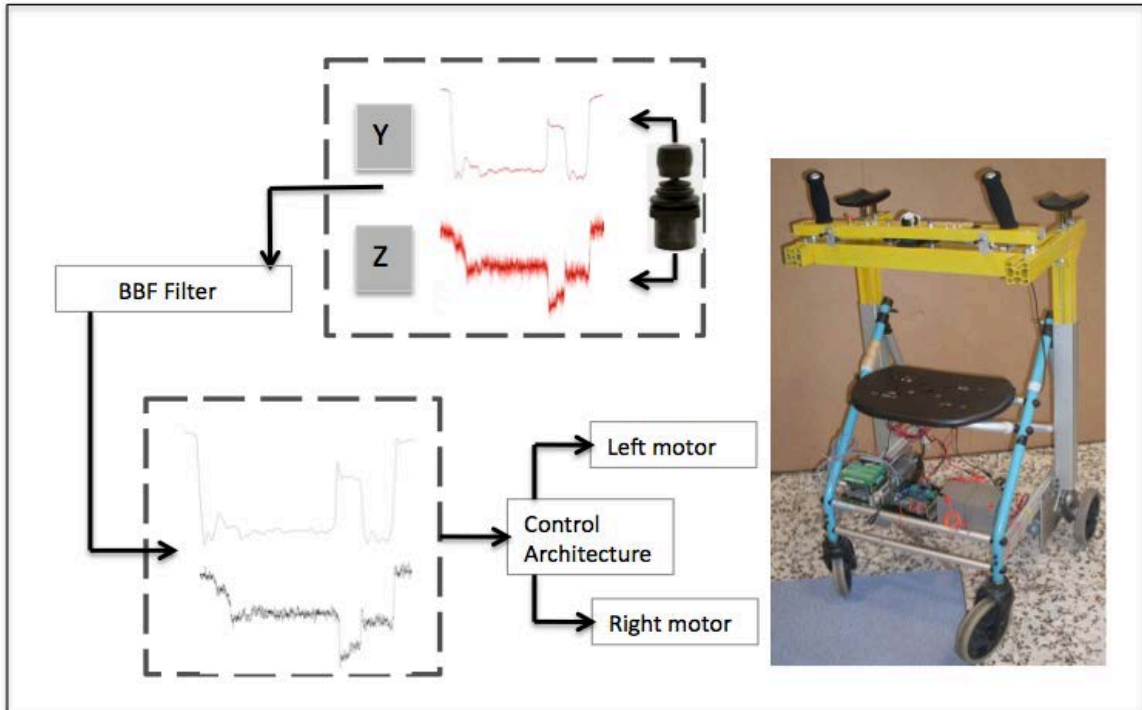


Figure 4.32 Complete system architecture implemented on NeoAsas walker.

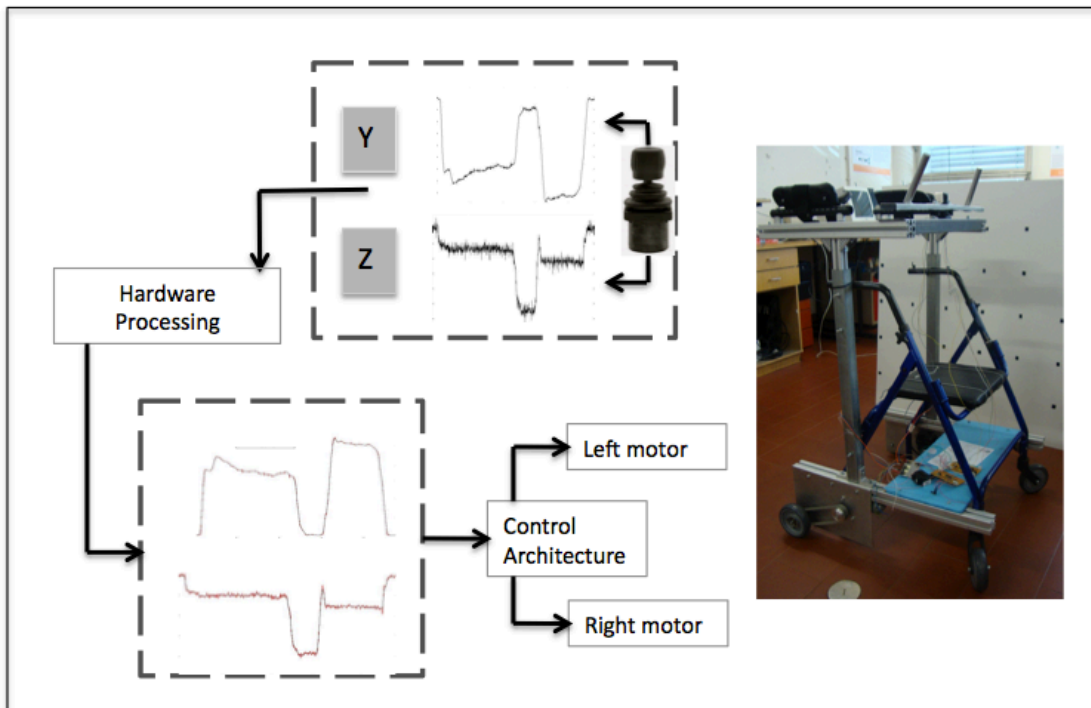


Figure 4.33 Complete system architecture implemented on ASBGo walker.

5. Fall and Grasp Detection – Safety considerations

5.1. Introduction

Falls are very common in elderly people due to several reasons such as osteoporosis or reduced protective reflexes. This has several consequences. Considering the users of and walking assistive device, a fall can lead to loss of confidence on the device.

Therefore the detection of possible falls of the user is one of the concerns and aims to integrate in the ASBGo walker. Additional safety is envisaged by several sensorial subsystems that complement each other. Two types of fall will be tackled: backward and forward fall.

A proposed subsystem composed by infrared sensors will improve the detection of falling forward situations during gait and stop the walker in time accordingly if required.

To detect if the user is falling backwards three subsystems were considered: (1) the walker cannot move backwards. So, if the user pushes the handles in his direction, the walker stops; (2) Another subsystem, based on two sensors, one on each forearm support, is intended to verify if the user is with his forearms properly supported on the base supports; (3) Two sensors placed on the handles ensure the user is guiding the walker grasping the two handlebars. If the two handlebars are not grabbed, two situations may occur: if the gait is starting, the walker will not move; if the user is already walking with the device, the walker will immediately stop.

These subsystems will ensure that a user fall is detected, stopping immediately the walker, not leaving it to roll away from the user while his falling. Thus, the walker will be either a support for the fall, or even preventing it from happening.

The sensors' choice was based in their simplicity, low cost and performance for this application.

5.2. Detection of a forward fall

Two areas were analysed for the best detection of a forward fall: (1) the lower base of the walker, at the height of the user's feet and; (2) the upper base of the walker, at the height of the user's chest.

5.2.1. Walker's lower base

The idea is to integrate a sensor in the lower base of the walker, and detect an area in which the lower limbs of the user are always present, when he is walking with the walker. If the user leaves this area it means that he is possibly falling.

For this purpose, the position of user's lower limbs related to the walker was analysed.

Firstly, it is necessary an analysis and a posteriori reconstruction of the movement of the lower limbs related to the structure of the walker in three dimensions. A movement analysis laboratory developed by BTS Bioengineering that has six CCD cameras sensible to the infrared radiation with its respective IR illuminators was used. Thus, 4 markers were placed in the walker and 4 on the user's lower limbs, displaced as figure 5.1 shows, to be detected by the BTS.

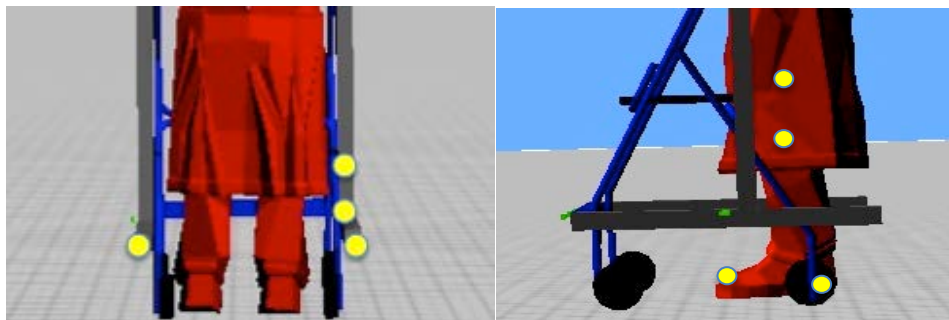


Figure 5.1 a) Sketch of the walker with the markers and b) the lower limbs of the user with the markers.

The trajectories, presented in table 4.1, were performed by three subjects, three times each, in the movement analysis laboratory and the results were analyzed in Matlab (Figure 5.2). In these figures it is represented the legs of the user and with the L form is represented the lower base of the walker.

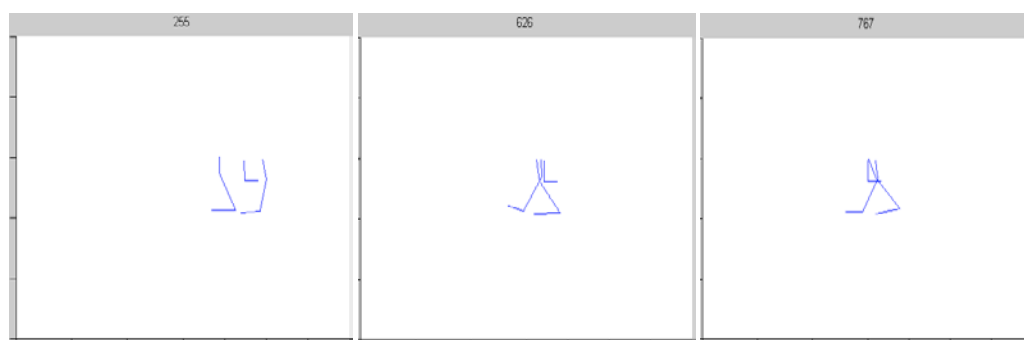


Figure 5.2 Frames from the BTS shootings. The lower base of the walker and the lower limbs in 2D.

After an extensive observation and analysis of the shootings taken by BTS, it was verified that there are several areas in which a user could position his lower limbs, as well as variability of walking forms. This fact difficult the definition of a specific area in which a leg and/or foot are always present. In figure 5.2, it is possible to verify the variability of positions that the lower limbs can present. The lower limbs, related to the walker, are not always in the same place and if the user falls forward, it is difficult to define the area that they will cross in that situation.

Therefore, it was determined that this was not a good solution to detect a forward fall.

5.2.2. Walker's upper base

The walker's upper base is at the height of the user's chest, and it was observed that if the user is falling forward, the distance between the user's chest and the walker decreases, and if not, this distance is maintained more or less constant. Thus, in order to measure this distance it is required a sensor in the upper base.

Firstly it was chosen a sensor to be integrated in the walker, to then perform experiments.

5.2.2.1. Sensor to detect the distance between the user's chest and the walker

Since this work has a commercial focus, the sensor selected has to be robust, low cost and require little electronic and power.

A laser solution is too expensive given the cost requirements. A solution using camera with structured light is possible to design for a reasonable price. It will reduce the amount of calculations for the processor depending on the structure of the light and therefore also the time it takes to implement a working system. The problem with this system is that different surfaces reflect different amounts of light. More calculations need to be added to determine where the person is if the camera cannot clearly detect it. Sunlight could also be a problem because it contains all wavelengths, and therefore could interfere with a structured light transmitter.

Infrared (IR) distance sensors are very low price and have a narrow beam width, which makes it suitable for this proposal.

A GP2Y0A21YK0F, in figure 5.3 (specifications in appendix B), from Sharp has been selected. It detects a distance from 7 cm to 80 cm with an analogue output, and has 5 ms of delay response.

This detector is quite inexpensive, use very little power, doesn't require external circuit, and fit in small spaces.



Figure 5.3 Sharp IR sensor (GP2Y0A21YK0F)

For the proposal presented here, to detect a forward fall, this sensor fits well, and it was chosen to be integrated on the walker.

5.2.2.2. *Detection of a forward fall*

The IR sensor was placed in the upper base of the walker, as it is shown in figure 5.4.

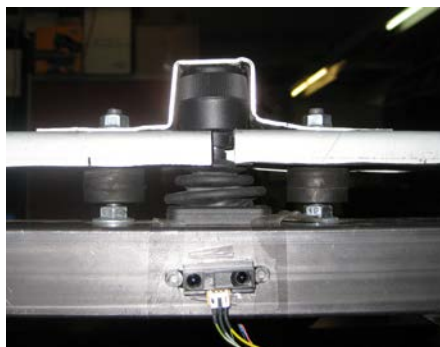


Figure 5.4 Sharp IR sensor integrated in the upper base of the ASBGo walker.

The trajectories, presented in Table 4.1, were performed by five subjects, three times each, and the output responses of the IR sensor were acquired using the Arduino platform. These experiments were performed to try to see if the user's distance related to the walker has a great variation, or not, and if it can be possible to define a "normal distance area" where the user is present when he is walking safely, and detect abnormalities when the user falls forward.

A typical signal for each trajectory is presented in figure 5.5.

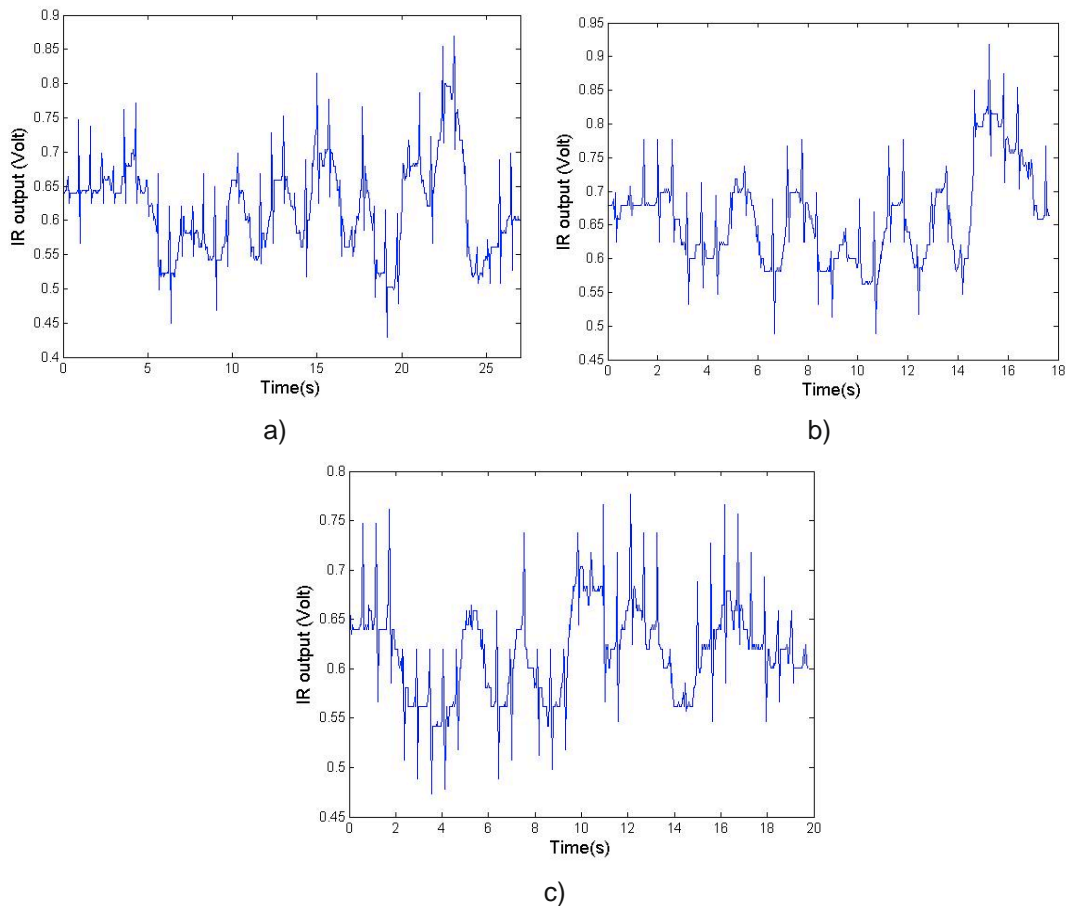


Figure 5.5 Typical IR output signal in trajectory a) A, b) B and c) C.

Looking at all the graphics, it is possible to observe that all begin in the same value, meaning that the person's distance to the sensor does not vary from usage to usage. This is ensured with the forearm supports that impose the person to be at a fixed distance relative to the walker.

The signals show a small oscillating behaviour, demonstrating that the user has a little oscillation of his trunk when is moving with the walker.

In figure 5.5a, it is denoted that the user approximates to the walker when turning right, between $t=20-25$ s, and the same happens in figure 5.5b, between $t=14-17$ s. However, its amplitude has not a significant difference in comparison with the rest of the signal's amplitude, presenting a variation of approximately 0.2V.

Thus, one can conclude that there is not a significance variation on the IR signal when the user is walking with the walker, besides the oscillations. And this was verified in all the experiments with the five users. Therefore, it can be defined a range of normal distance between the user and the walker, during a safety usage.

Figure 5.6 shows a typical signal of a user that is walking normally, and then starts to fall forward at $t=18-19$ s. As it was expected, the amplitude of the signal increased, since the distance between the user and the walker decreased. In comparison with the rest of the signal, this increase, in the order of 1V, is very significant and can be easily detected.

Thus, it is possible to conclude that, with this sensor, a dangerous situation can be detected.

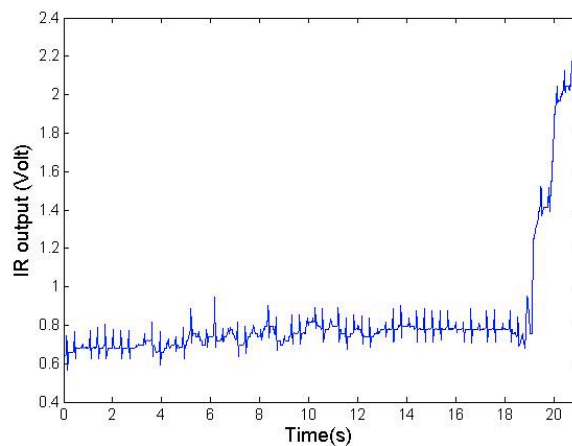


Figure 5.6 Typical IR output signal from user that is walking normally and then falls forward.

The option of using this sensor to detect also backward falls was analysed. However, it was concluded that if a person increases his distance from the sensor, the signal amplitude variation is very small, not being suitable for a great detection of backward falls. This conclusion was taken with base on the output response of the IR sensor for high distances. As it can be seen in Appendix B.2, after the 50 cm, the variation of the signal is very small, and hardly detected.

An algorithm was proposed to detect the variation of the signal, informing the walker if the user was walking normally, or falling. The algorithm is presented in figure 5.7.

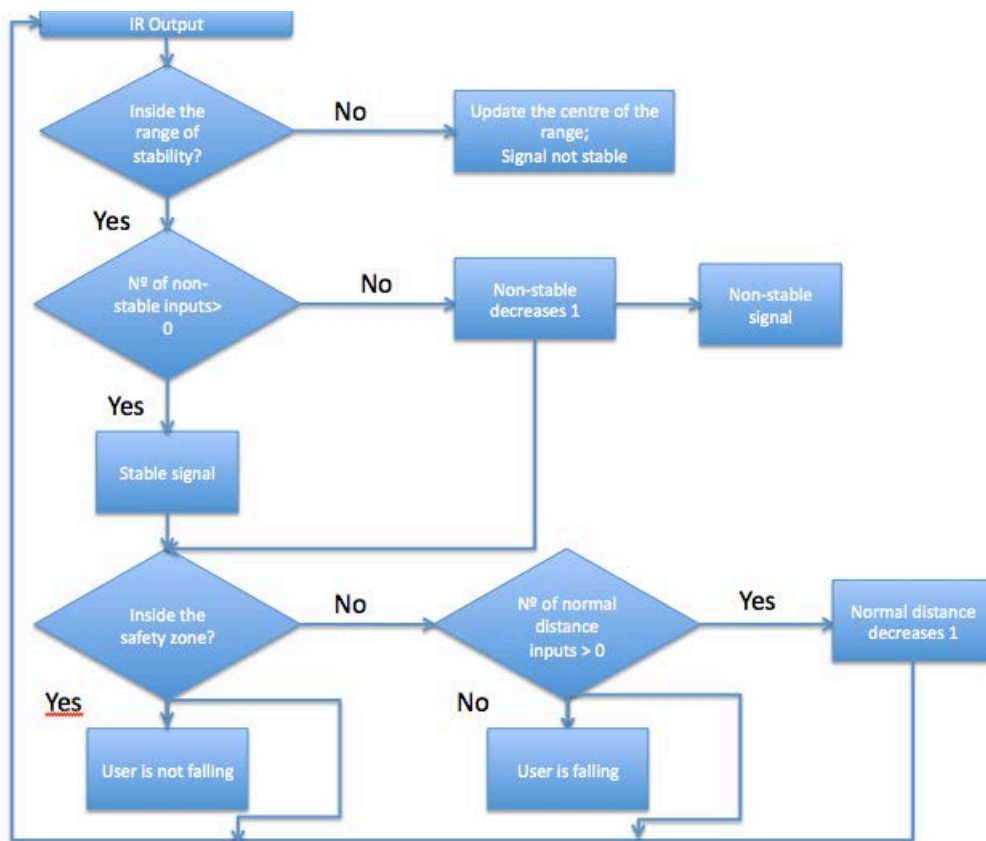


Figure 5.7 Algorithm to detect if the user is falling or not through the readings of the IR sensor output signals.

The algorithm input is the IR signal output. This signal is evaluated in terms of its behaviour, *i.e.* if it varies a lot or not, according to a non-stable variable. This is done to identify the behaviour of the signal, for example, here it can be observed the user's oscillations, while he is walking. If the signal is stable, *i.e.* its value varies between an a priori defined stability range, the variable non-stable decreases, if not, the centre of the range of stability is updated following the signal's behaviour, *i.e.* its value is now equal to the input at that instant.

Then, if the signal is stable, it is evaluated why this happened: the user is stopped and he is not oscillating his trunk or he is falling forward. So, first it is verified if the signal is around an a priori defined "safety zone". If the user is within this zone, the walker is informed that the user is safe. If the user is out of the safety zone, it means the user might be falling forwards. First some samples are acquired and the variable "nº of normal distance in inputs" is decreased. If this variable reaches zero, the walker stops, because the user is falling. If not, it was a false alarm due to noise.

Thus, the algorithm will detect if the output signal of the IR signal is around zero, inside an a priori defined safety range or if it exceeds the defined range, detecting a danger situation.

This algorithm has the advantage of following near the behaviour of the signal, and does not require an extra filter to deal with noise and can thus detect false alarms.

Beyond the ability to detect if the user is falling, this algorithm has the potential to analyse the postural stability of the user and this can be used in the identification of patterns of trunk oscillations. This will be used in future works for identification of walking pathologies, interaction user-walker, etc.

5.3. Detection of a backwards fall.

To detect if the user is falling backwards three subsystems were considered: (1) the walker cannot move backwards. So, if the user pushes the handles in his direction, the walker stops; (2) Another subsystem, based on two sensors, one on each forearm support, is intended to verify if the user is with his forearms properly supported on the base supports; (3) Two sensors placed on the handles ensure the user is guiding the walker grasping the two handlebars. If the two handlebars are not grabbed, two situations may occur: if the gait is starting, the walker will not move; if the user is already walking with the device, the walker will immediately stop.

Subsystem (1) will be not described in this chapter, since it will be integrated in the control strategy, presented in Chapter 6. The other two subsystems will be presented and explained in detail in the following sections.

5.3.1. Detection of contact on the forearm supports

The user has to guide the walker with his forearms properly relied on the supports, to have a suitable sustain. These supports can eliminate the degree of freedom of the elbow articulation and a higher fraction of the user's weight is supported by the device, making it easier to push and increasing the friction component of the system, reducing the risks of glide.

However, if the user falls backward he will release the forearm's support. So, it is needed to make sure that the walker detects this situation and stop, accordingly. This is done by placing one sensor on each forearm support.

5.3.1.1. Sensor to detect user's contact with the forearm supports

Initially the idea was to place IR sensors on the supports, since they are low cost and small, fitting inside the support. So, an IR sensor (Sharp-GP2Y0A41SK0F) with a detection range of [0.6, 10] cm was integrated on the supports and some experiments were run.

After some experiments, it was concluded that this was not a good option because if the user was not in contact with the support, but still close ($\approx 3\text{cm}$), the sensor indicates that the user is still in contact with the support did not respond fast enough. This occurs mainly because of two reasons. First, the output response of the sensor for a distance of 6 mm and 3 cm is equal. Second, the structure of the support influences on the IR light reflection and causes a wrong reading of the distance between the user's forearm and the support.

Thus, IR sensors were replaced by force sensors, since they can detect if any weight is being supported by the supports. These sensors have to be small or thin, low cost and present a fast response.

The *FlexiForce* sensors (Figure 5.8) fulfilled the criteria. The *FlexiForce* sensor is an ultra-thin and flexible printed circuit, which can be easily integrated into the forearm supports. With its paper-thin construction, flexibility and force measurement ability, the *FlexiForce* force sensor can measure force between almost any two surfaces.



Figure 5.8 *FlexiForce* sensor.

The sensors are constructed of two layers of substrate, composed of polyester film. On each layer, a conductive material (silver) is applied, followed by a layer of pressure-sensitive ink. Adhesive is then used to laminate the two layers of substrate together to form the sensor. The silver circle on top of the pressure-sensitive ink defines the “active sensing area”. Silver extends from the sensing area to the connectors at the other end of the sensor, forming the conductive leads. More details in appendix C.1.

FlexiForce sensors are terminated with a solderable male square pin connector, which allows them to be incorporated into a circuit. The two outer pins of the connector are active and the centre pin is inactive.

The sensor acts as a variable resistor in an electrical circuit. When the sensor is unloaded, its resistance is very high (greater than 5 Meg-ohm); when a force is applied to the sensor, the resistance decreases. The implemented circuit is presented in appendix C.2.

The sensing area of this sensor is only 9.5 mm of diameter. However, the user applies load in the entire support. This may lead to wrong detections or missing detections. In order to deal with this problem a mechanical device, depicted in figure 5.9, was developed for the physical integration of the *FlexiForce* sensor in the support forearms. The mechanical device is consisted on two metal plated with a rubber cube, smaller than the sensing area, placed above the sensing area of the sensor, to ensure that the entire load that is applied by the user on the support goes to this area. So, when the user supports on the walker, the device will close, and the rubber will do a perpendicular pressure in the sensing area, transmitting to the walker the information that the user is properly supported.



Figure 5.9 Mechanical device to support the sensor and ensure that the load made by the user on the forearm support is entirely made on the sensing area of the *Flexiforce* sensor.

Finally, the assembly of this mechanism on the walker is presented in figure 5.10.



Figure 5.10 Assembly of the *FlexiForce* sensor with the forearm support.

A typical output from the *FlexiForce* is presented in Figure 5. 11.

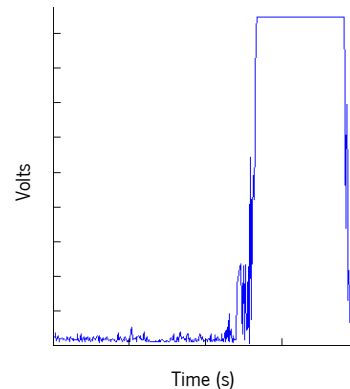


Figure 5.11 Typical signal of the *Flexiforce* sensor when first, the user is not loading the sensor, and then loads.

When the user is not loading the sensor the output signal is zero. When the user relies on the support, the sensor immediately responds to the force, and the signal increases. When the user leaves the support, the sensor's signals immediately decreases.

In this application, the main aim is to detect if the user is or is not relied in the forearm support. Hence, the acquisition electronic was designed in a way so that the sensor responds quickly to the load of the user and saturates easily. This saturation is useful because in this way, the signal is stable and is not influenced by possible oscillations of the user's trunk.

An algorithm was developed to read the signal and identify when the user was not loading the sensor, meaning that he is falling, or not properly supported on the supports.

This algorithm is very similar to the one showed in figure 5.7, and is presented below:

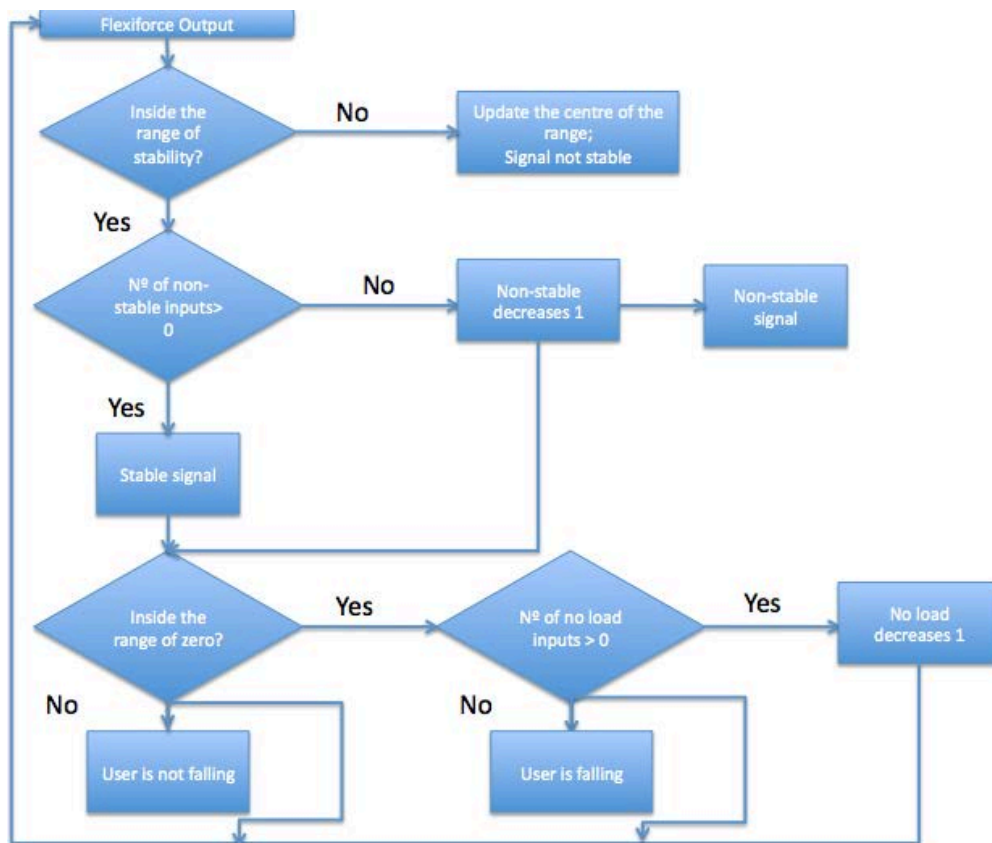


Figure 5.12 Algorithm to detect if the user is falling or not through the readings of the flexiforce output signals.

With this algorithm, it is possible to detect whether there is a load, as well as, evaluate the stability of the support. Similarity to the algorithm used to detect backwards falls, here it is not needed a filter to attenuate the noise, and thus no additional computational cost is added to the final algorithm to control the walker’s movement. This fact is relevant when discussing real-time applications.

The algorithm input is the force sensor signal output. The signal is evaluated in terms of its behaviour considering if it is stable, it is zero or if it is different from zero.

If the signal varies, the algorithm follows its behaviour. But, if the signal is stable, i.e. varies between an a priori defined range of stability, two situations could be happening: or the user is safely supported, or he is falling. If the user is safely supported, the signal is around zero, and the walker is informed that the user is safe. On the other hand, if the user is falling, it is detected that the user is out of the pre-defined “range of zero”. Thus, some samples are acquired and the

variable “n° of load inputs” is decreased. If this variable reaches zero, the walker stops, because the user is falling. If not, it was a false alarm due to noise.

For the moment, it was not considered the absolute value of the data measured by the force sensor. But this will be considered in future studies, to quantify the stability that the forearm supports provide to the user, as well as, identify asymmetries on the gait of the user, and other pathologies.

This algorithm is integrated onto the Arduino platform code and is part of the walker’s movement controller.

5.3.2. Detection of handles grasping.

Another way to detect if the user is falling backwards is to verify if he is grasping properly the handles of the upper base of the walker. So, by placing sensors on the handles, one can detect if he is in contact or not with the handles.

Since the previous study demonstrated that the *FlexiForce* sensors suit well to detect contact/force interactions, they were chosen similarly to also detect if the user is grasping the handles.

Other approaches were considered, such as the use of capacitor sensors, piezoelectric sensors, IR sensors and buttons. All these were experienced and studied and shown to be suitable options. However, as the author of this thesis has the goal in the future to study the interaction between the user intentions and the movement commands transmitted through the handles, the *FlexiForce* sensors were the ones chosen, since the others just suit for an on/off control.

The same algorithm, mechanism and electronics, applied in the detection of backwards fall were used as well for this detection. Each handle has its own algorithm for detection, *i.e.* if the user releases just one handle, the walker stops.

5.4. Conclusions

In this chapter were explained the different safety mechanisms considered in the developed walker.

To identify if the user is falling forward, an IR sensor was integrated in the upper base of the walker structure. This sensor measures the distance between the user and the walker,

successfully, and with the help of a developed algorithm, the walker is advised to stop when the distance between the user and the walker decreases significantly.

To identify if the user is falling backwards, *FlexiForce* sensors were integrated in the forearms support as well as in the handles of the upper base of the walker structure. This sensor measures the force that is made by the user when he relies onto the supports; and the force that he makes when grasping the handles. A proposed algorithm advises the walker to stop when no force is applied onto the support and/or in each handle.

The proposed algorithm has been programmed in the Arduino platform combined with the processing of the joystick's signals (Chapter 4) and the control strategy to control the motors (Chapter 6).

6. Control Strategy of the walker's motors

6.1. Introduction

This work aims to develop a Smart Walker that is driven intelligently, through the acquisition and processing of signals from a joystick sensor when the user is walking with the Smart Walker. With the joystick, it is possible to infer the user's command intentions considering direction and velocity.

Thus, in this Chapter it is addressed a control strategy to classify the signals sent by the joystick and transform them into motor outputs (direction and velocity), in such way that the walker drives the motors according to the user's commands. Using two DC motors, the rear wheels of the Smart Walker were driven according to the commands of the user sent through the joystick.

In this chapter, it was tackled the overall assembly of the sensors, the processing platform, the motors and the auxiliary circuits. This assembly took into account the mechanical stability of the walker, putting the heavier elements at the lowest part of the walker.

6.2. Mechanical structure of the motor control unit

To start the experiments with the motors, it was used a pair of DC motors connected to a controller designed to perform as an interface between the control strategy and the motors. This pair of motors is 24 VDC powered, has a rate speed of 40 rpm, nominal current of 2.5 A and nominal torque of 2.5 Nm (Appendix D.1). They also include a self-locking system to improve the safety of the users in case of any technical fault or in the absence of sufficient electrical power, caused by battery discharge, for example.

The motor controller is included in a circuit board that is also prepared to monitor voltage current or temperature problems, as well as short-circuits.

Each motor is installed next to a rear wheel of the walker, leaving the two front wheels free to move omni-directionally. A mechanical connection between the wheel and the motor for both walkers (NeoASAS and ASBGo) is provided by a chain, as it is shown in figure 6.1. Since there is a

necessity of working with the walker without the motors, the wheels were not directly connected to the motor. Thus, it is always possible to perform certain experiments without any motorization.

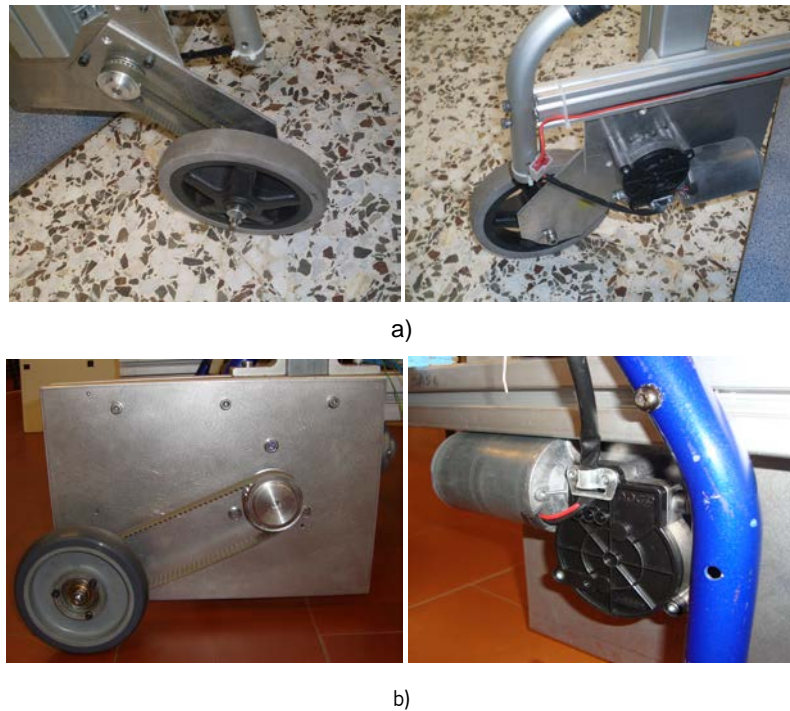
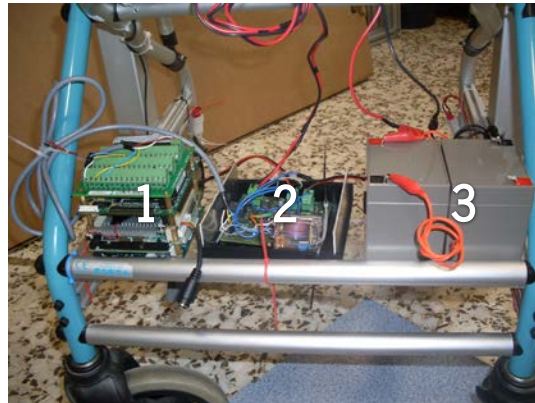


Figure 6.1 Motors on walker's structure of a) NeoASAS and b) ASBGo walkers.

Considering the NeoASAS walker, the connection between the motors and the sensors is performed through a PC/104, which contains the control strategy developed to guide the walker. The inputs of the system (signals from the sensors) are an analog voltage between 0 and 5 V. The PC/104 analog outputs are between 2.5V and 5V and connect to the control board of the motors (Appendix D.2) that contains a microcontroller and H-bridges, which convert this voltage to the range of 0-24V, to power the motors.

Three plumb batteries were used, one with 5 V and two with 12 V to power the control board, the sensors and the motors.

The overall assembly of the NeoASAS walker is shown in figure 6.2.

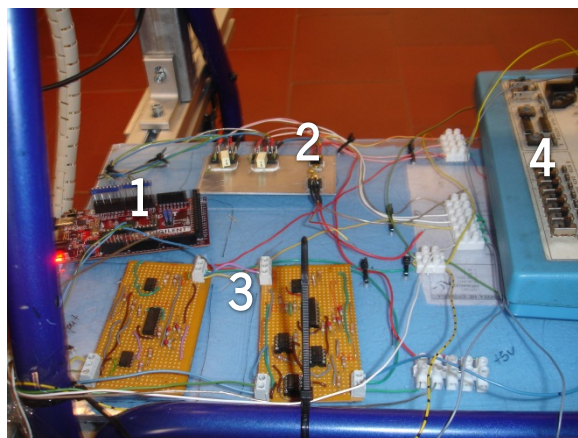


1	PC/104
2	Board control
3	Batteries

Figure 6.2 Processing platform, Board control and batteries located on the lower zone of the NeoASAS walker.

Considering the ASBGo, the connection of the sensors and their auxiliary circuits with the motors is performed through an Arduino platform (Chipkit MAX32), where the control strategy was programmed. The Arduino platform also works as a control board that is connected to H-bridges (Appendix D.3) that convert the PWM signals sent by the Arduino platform to the motors as a range of 0-24V.

A digital lab as a power source of 12 V and -12 V, a positive voltage regulator (7805) to convert 12V into 5V and a negative voltage regulator to convert -12 V into -5V power the overall system.



1	Arduino
2	H-bridges
3	Auxiliary circuits
4	Power Source

Figure 6.3 Arduino Platform, H-bridges, sensors' auxiliary circuits and the power source.

6.3. Design and implementation of a control strategy to drive the walker

Intelligent systems have shown many advantages in the control and modelling of engineering system. They have the ability to exhibit advantages in performance over more conventional mathematical techniques by rapidly modelling and learning characteristics of multi-variant complex systems. [106].

Fuzzy logic is a problem-solving technique that derives its power from its ability to draw conclusions and generate responses based on vague, ambiguous, incomplete and imprecise information. To simulate this process of human reasoning it applies the mathematical theory of fuzzy sets [107].

This algorithm is based on the definition of rules that are evaluated in parallel and relate to input variables. This approach is consistent with the reasoning based on rules that define the problem of detection of user's command intentions.

It is also possible to incorporate an a priori knowledge in a direct manner in the model of the system, given the simple linguist approach.

The purpose detection of command intentions presents variability between observations and subjects, which requires a method that defines a flexible and adjustable system for each specific case. Knowledge about the user's command intentions is limited and there is no exact mathematical model that characterizes them.

Thus, the management of the interaction between the user and the walker is exactly the type of problem and issue for which a fuzzy logic approach appears to be applicable and has the potential for making better, quicker and more accurate predictions than classic methods.

6.3.1. Theory of the Fuzzy sets

The theory of fuzzy sets is an extension of classical set theory and it is, in turn, mathematically rigorous. Within the logic of classical sets, an element is or is not part of a set in question. The distinctions between membership or not membership to a particular element of a set based on a precise and rigorous border, assigned as 0 or 1, to indicate the membership of that set don't allow to consider a degree of partial membership of an element in a set. The theory of fuzzy sets on the other hand, accepts the definition of partial membership, extending the classical theory. An example is showed below to illustrate the concept:

Considering B a set of real numbers and,

$$B' = \{b \in B \mid b \text{ is very negative}\} \quad (6.1)$$

The subset B' is a fuzzy subset because is not accordingly defined with the classic logic. Then, if a membership function is introduced as a representation to measure the very negative property, a fuzzy set is defined. Figure 6.4 shows a membership function, which can be chosen for such representation. Here the b value corresponds to a membership value μ of a membership function μ_b that represents the very negative property.

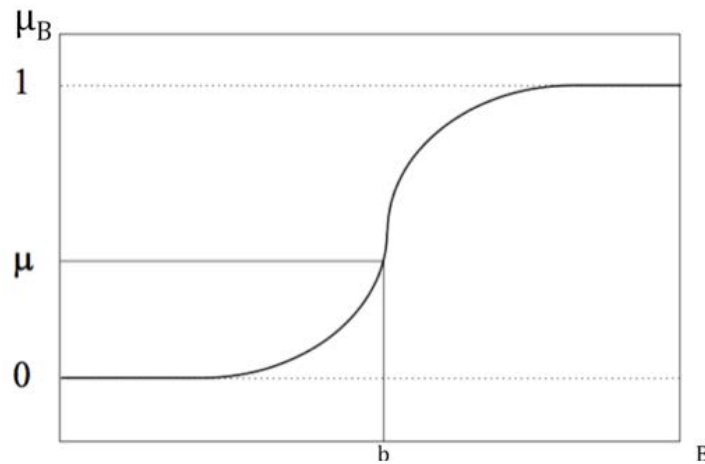


Figure 6.4 Membership function for a very negative value.

6.3.1.1. The Model

Obtaining a fuzzy model corresponds to an approximate reasoning rather than fixed and exact, which refers to the methodology for describing complex physical systems (non-linearity, randomness, uncertainty, and so on). Suppose that the system is known as a “black box”, which can measure only one set of inputs x_1, \dots, x_n and outputs z_1, \dots, z_m .

The model of the system consist, in general terms, in obtaining a mathematical description using functions, formulas or classic linguistic statements of the relationship between inputs and outputs. The model of the fuzzy system will consist of how to quantify linguistic logic by fuzzy logic and a functional mathematical model. The basic elements of the fuzzy system (Figure 6.5) are:

- The knowledge base (rules);
- An inference mechanism (Decision making logic);
- A fuzzification interface (fuzzifier);
- A defuzzification interface (defuzzifier);

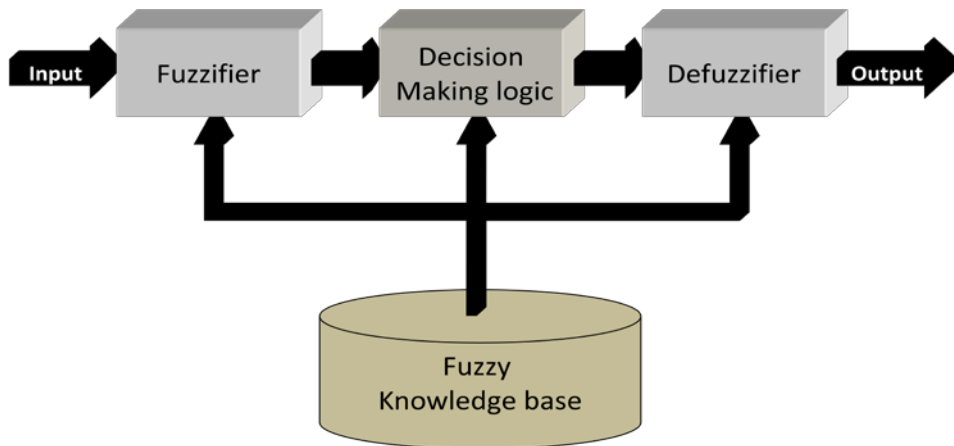


Figure 6.5 Fuzzy logic system diagram.

And the steps in the fuzzy reasoning are:

- **Fuzzification:** compares the input variables with membership functions (MF) of the premises (IF) to obtain the membership values, μ , between 0 and 1.
- **Inference:** consists in the application of fuzzy logic operators (AND, OR, etc.) in the membership values of the premises to obtain a single numerical value between 0 and 1.
- **Generation:** create the consequent part (THEN) from the fuzzy rule.
- **Defuzzification:** consists in decoding the information after the aggregation of the fuzzy rules and turn the output into a numerical value.

6.3.1.2. Fuzzification

Through the fuzzification process it will be found the membership values μ of the inputs x_n with the fuzzy sets. In table 6.1, the parameterized functions commonly used to define MF are presented. The MF parameters require that a knowledgeable human operator initialise them. The operator must optimise the parameter values to achieve a required level of accuracy of mapping of the physical system by the fuzzy system. The MF should overlap to allow smooth mapping of the system. Figure 6.5 depicts some examples of the representations of the MF.

Table 6.1 General Membership Funtions.

Membership Function (MF)	Equation	Parameters
Triangular MF	$\text{triangle}(x; a, b, c) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right)$	a, c (boundaries) and b (center)
Trapezoidal MF	$\text{trapezoid}(x; a, b, c, d) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right)$	a, d (boundaries) and b, c (centers)
Gaussian MF'	$\text{gaussian}(x; \sigma, c) = e^{-\left(\frac{x-c}{\sigma}\right)^2}$	c (mean) and σ (standard deviation)
Bell MF	$\text{bell}(x; a, b, c) = \frac{1}{1 + \left \frac{x-c}{a}\right ^{2b}}$	a, b (boundaries) and c (center)
S-shaped	$s(x; a, b) = \frac{1}{1 + e^{-a(x-b)}}$	a, b (boundaries)

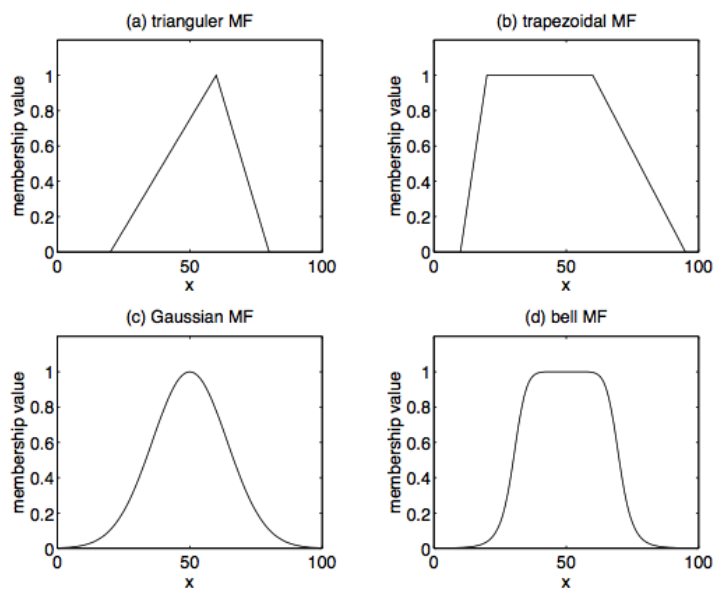


Figure 6.6 Examples of membership functions types [108].

6.3.1.3. Inference and Generation

The types of rules that can be defined for the design of the fuzzy model can be classified as [109]:

(1) Fuzzy rules with a constant output in the consequent:

$$R_i : \text{IF } x_1 \text{ is } A_i \text{ and } \dots \text{ and } x_n \text{ is } B_i, \text{ THEN } z_1 \text{ is } c_i \text{ and } \dots \text{ and } z_m \text{ is } c_w \quad (6.2)$$

(2) Fuzzy rules with a linear combination of the input variables in the consequent:

$$R_i : \text{IF } x_1 \text{ is } A_i \text{ and } \dots \text{ and } x_n \text{ is } B_i, \text{ THEN } z_1 \text{ is } g_i(x_1, \dots, x_n) = a_0 + a_1 x_1 + \dots + b_n x_n \text{ and } \dots \text{ and } z_m \text{ is } g_w(x_1, \dots, x_n) = d_0 + d_1 x_1 + \dots + d_n x_n \quad (6.3)$$

(3) Fuzzy rules with a fuzzy set in the consequent:

$$R_i : \text{IF } x_1 \text{ is } A_i \text{ and } \dots \text{ and } x_n \text{ is } B_i, \text{ THEN } z_1 \text{ is } C_i \text{ and } \dots \text{ and } z_m \text{ is } C_w \quad (6.4)$$

Where R_i is the i -nth fuzzy rule, x_m ($m=1,2,3,\dots,n$) are the inputs of the system that were introduced in the antecedent of the rule and y is the output, introduced in the consequent of the rule. A_i , B_i and C_w are the fuzzy set and c_w , b_n and d_n are constants.

The approach (3) was the one used in this work.

The next stages of these steps are represented in figure 6.7.

After the definition of each fuzzy rule, a single numerical value between 0 and 1 is found for each antecedent. This numerical value C'_i is the minimum μ between the inputs of the corresponding antecedent.

Then each C'_i will be the membership value of the consequent fuzzy set C_w corresponding to an output z (belongs to $[z_{\min}, z_{\max}]$) of this set.

The aggregation of the rules is done through the calculation of the maximum, *i.e.* For each output z found, its corresponding output membership value will be the maximum between the C'_i of each R_i . This maximum will be C' .

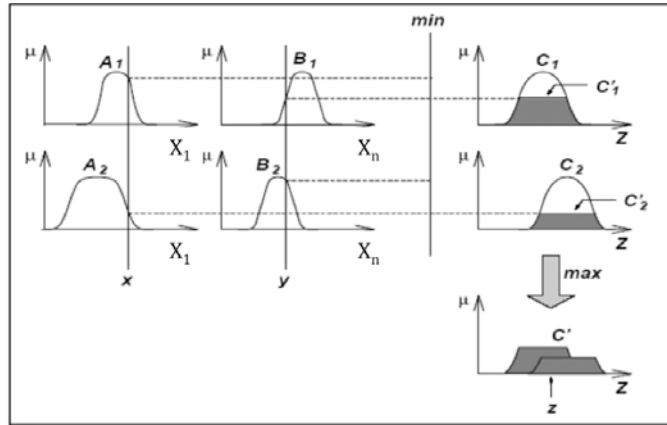


Figure 6.7 Inference and Generation steps of the Fuzzy logic system [108].

6.3.1.4. Defuzzification

Defuzzification is such inverse transformation that maps the output from the fuzzy domain back into the numerical domain. As a defuzzifier, the combination of the results of the previous step can be done by different strategies. These strategies are depicted in figure 6.8.

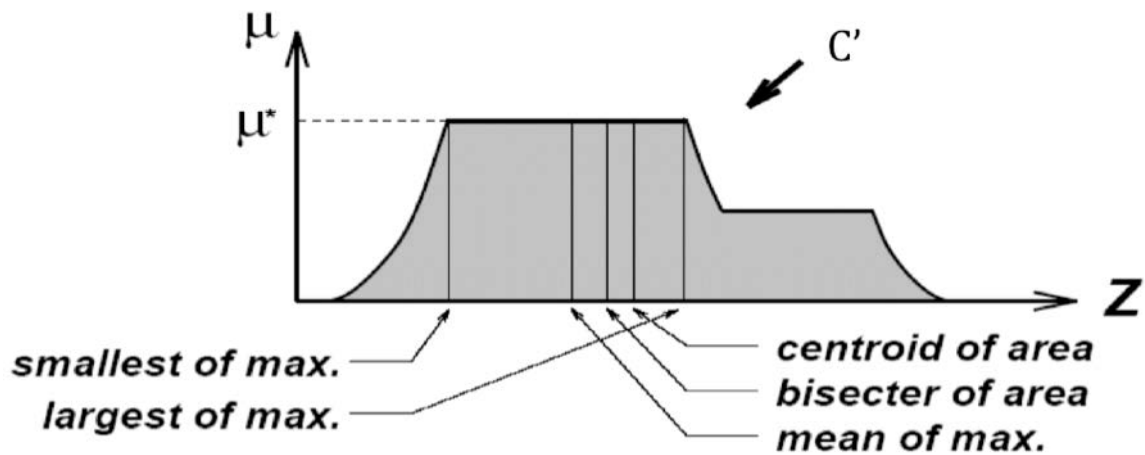


Figure 6.8 Various defuzzification schemes for obtaining a numerical output [108].

The most frequently used defuzzifier is the centroid of area (COA) [110] and considering the results from Rotshteina, this method is the one with higher precision in comparison with the other methods. It computes the centroid of the composite area representing the output fuzzy term and is defined as:

$$z_{COA} = \frac{\int \mu_{C'}(z).zdz}{\int_z \mu_{C'}(z)dz} \quad (6.5)$$

where $\mu_{C'}(z)$ is the aggregated output MF.

6.3.2. Design of the Fuzzy Inference System to control the walker's motors

In this proposal the system has two inputs and two outputs. The inputs of the system correspond to the signals of forward (Y) and rotation (Z) of the joystick, that were presented in chapter 4, and the outputs correspond to the signals that will be sent to the pair of motors, MR and ML.

The inputs are in the interval [-1,1] and the output values belong to [0, 1], so that no negative outputs for activation of the motors. Thus, the walker cannot move backwards, not pushing the user backwards, which reduces the risk of causing falls.

Considering the NeoAsas walker, the fuzzy system was implemented in Matlab using the *Fuzzy Logic Toolbox*. Considering the ASBGo walker, it was implemented in C/C++ into the Arduino platform.

The first step to take is to quantify the MF accordingly with a criterion based on the analysis of the user's command intentions, made in Chapter 4.

Four MFs were defined to Y-signal and five MFs were defined to the Z-signal. Also, three MFs were defined for the two outputs. Each MF of the inputs considers a type of velocity and direction and try to cover all the information that is transmitted by joystick. With the Y-signal one can have four types of information for the velocity and direction: Negative (Neg) informs to go backwards, Zero means stopped, little positive (LP) is related to go forward but slowly and very positive (VP) is also related to go forward but fast. Z-signal transmits five types of information for the velocity and direction: much left (ML) is related to turn fast to left, little left (LL) is related to turn slowly to left, Zero means stopped, little right (LR) is related to turn slowly to right and much right (MR) is related to turn fast to right.

The output's MFs try to cover the three possibilities for the two motors: Be stopped (Zero), go fast (F) and go slow (S). To turn left the right motor will have a faster velocity than the left one, and to turn right the left motor will have a faster velocity than the right one.

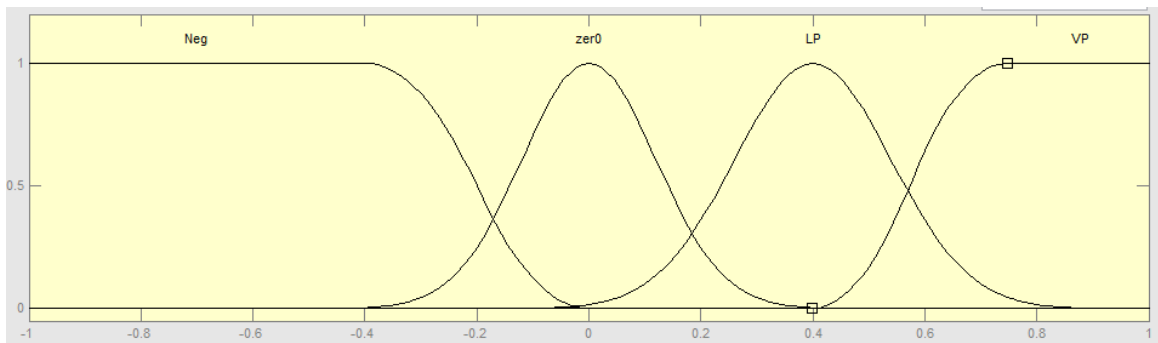
For the inputs and outputs two types of membership functions were defined: Gaussian and S-shaped MF's (Table 6.1).

Due to triangular and trapezoidal simple formulas and computational cost efficiency, both MF's have been extensively used, especially in real-time applications [108]. However since the MF's are composed of straight-line segments, they are not smooth at the switching points specified by the parameters. Gaussian and S-shaped are smoother and with concise notation. So they were more suitable for this application, due to their smoothness.

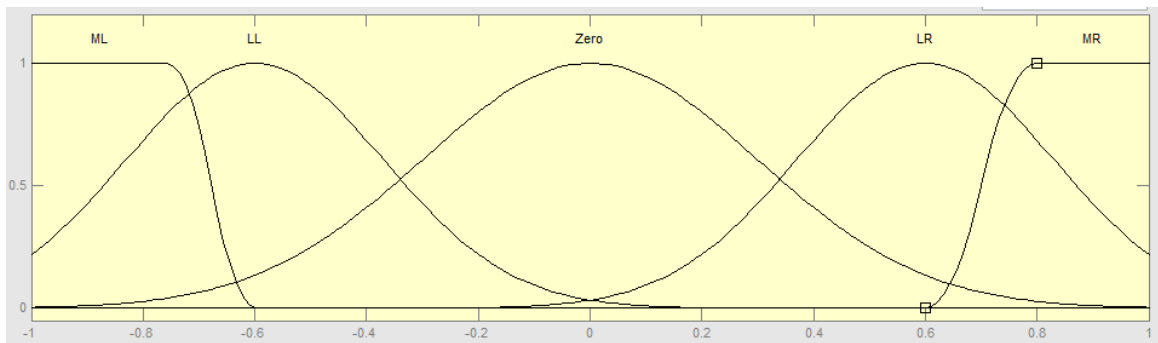
After defining these membership functions, the name and parameters values necessary to the inputs and outputs of fuzzy control system were tuned manually. These are shown in table 6.2 and the respective graphs are shown in figure 6.9.

Table 6.2 Membership functions of the inputs/outputs of the fuzzy inference system.

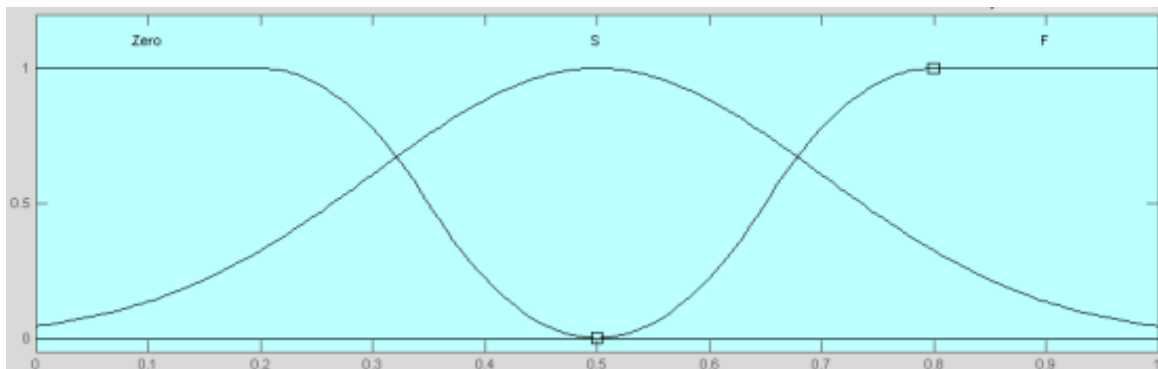
Input/Output	Signal	Name of membership function	Membership function type	Parameters values
Input	Forward (Y)	Negative (Neg)	S-shaped	$a=-1; b=0$
		Zero	Gaussian	$c=0; \sigma=0.12$
		Little Positive (LP)		$c=-0.4; \sigma=0.15$
		Very Positive (VP)	S-shaped	$a=0.4; b=1$
	Rotation (Z)	Much Left (MR)	S-shaped	$a=-1; b=-0.6$
		Little Left (LL)	Gaussian	$c=-0.6; \sigma=0.23$
		Zero		$c=0; \sigma=0.3$
		Little Right (LR)	$c=0.6; \sigma=0.23$	
Output	Left and righth motor (MR and ML)	Stopped (Zero)	S-shaped	$a=0; b=0.5$
		Slow (S)	Gaussian	$c=0.5; \sigma=0.2$
		Fast (F)	S-shaped	$a=0.5; b=1$



a)



b)



c)

Figure 6.9 Membership functions for a) the forward input (Y), b) the rotation input (Z), and c) Motors. X-axis represents the numerical values of the inputs/outputs, and the Y-axis represents the degree of membership μ . These images are a print screen of the functions generated in Matlab.

The next step is to define the rule base that the system will operate under. There were defined 20 rules for the movement of the walker. These rules are the combination of the two inputs (one input has four MFs and the other has five MFs) and the two outputs ($5 \times 4 = 20$ combinations). The rules are:

1. **IF** rotation = ML **AND** forward = Neg **THEN** ML = Zero **AND** MR = S;
2. **IF** rotation = ML **AND** forward = Zero **THEN** ML = Zero **AND** MR = F;
3. **IF** rotation = ML **AND** forward = LP **THEN** ML = S **AND** MR = F;
4. **IF** rotation = ML **AND** forward = VP **THEN** ML = S **AND** MR = F;
5. **IF** rotation = MR **AND** forward = Neg **THEN** ML = S **AND** MR = Zero;
6. **IF** rotation = MR **AND** forward = Zero **THEN** ML = F **AND** MR = Zero;
7. **IF** rotation = MR **AND** forward = LP **THEN** ML = F **AND** MR = S;
8. **IF** rotation = MR **AND** forward = VP **THEN** ML = F **AND** MR = S;
9. **IF** rotation = LL **AND** forward = Neg **THEN** ML = Zero **AND** MR = Zero;
10. **IF** rotation = LL **AND** forward = Zero **THEN** ML = Zero **AND** MR = S;
11. **IF** rotation = LL **AND** forward = LP **THEN** ML = S **AND** MR = F;
12. **IF** rotation = LL **AND** forward = VP **THEN** ML = S **AND** MR = F;
13. **IF** rotation = LR **AND** forward = Neg **THEN** ML = Zero **AND** MR = Zero;
14. **IF** rotation = LR **AND** forward = Zero **THEN** ML = S **AND** MR = Zero;
15. **IF** rotation = LR **AND** forward = LP **THEN** ML = F **AND** MR = S;
16. **IF** rotation = LR **AND** forward = VP **THEN** ML = F **AND** MR = S;
17. **IF** rotation = Zero **AND** forward = Neg **THEN** ML = Zero **AND** MR = Zero;
18. **IF** rotation = Zero **AND** forward = Zero **THEN** ML = Zero **AND** MR = Zero;
19. **IF** rotation = Zero **AND** forward = LP **THEN** ML = S **AND** MR = S;
20. **IF** rotation = Zero **AND** forward = VP **THEN** ML = F **AND** MR = F;

By this, it is built the model for the discrimination between different types of user's commands to guide the walker.

A scheme that depicts the behaviour of the fuzzy system while the user is guiding the walker is shown in figures 6.10 and 6.11. In figure 6.10 the user is performing trajectory A and in figure 6.11 he is performing trajectory B (Chapter 4). In these schemes are represented: sequence of snap shots of a user walking with the device while is performing the pre-defined trajectories (A and B); fuzzification of the outputs of the joystick (Y and Z-signals); and finally the representation of the inference and defuzzification that have as a result the centroid of area of the output MFs' for each motor (ML-C and MR-C). These outputs will be converted to pre-set intervals in order to power the motors, so that they can follow the user's commands.

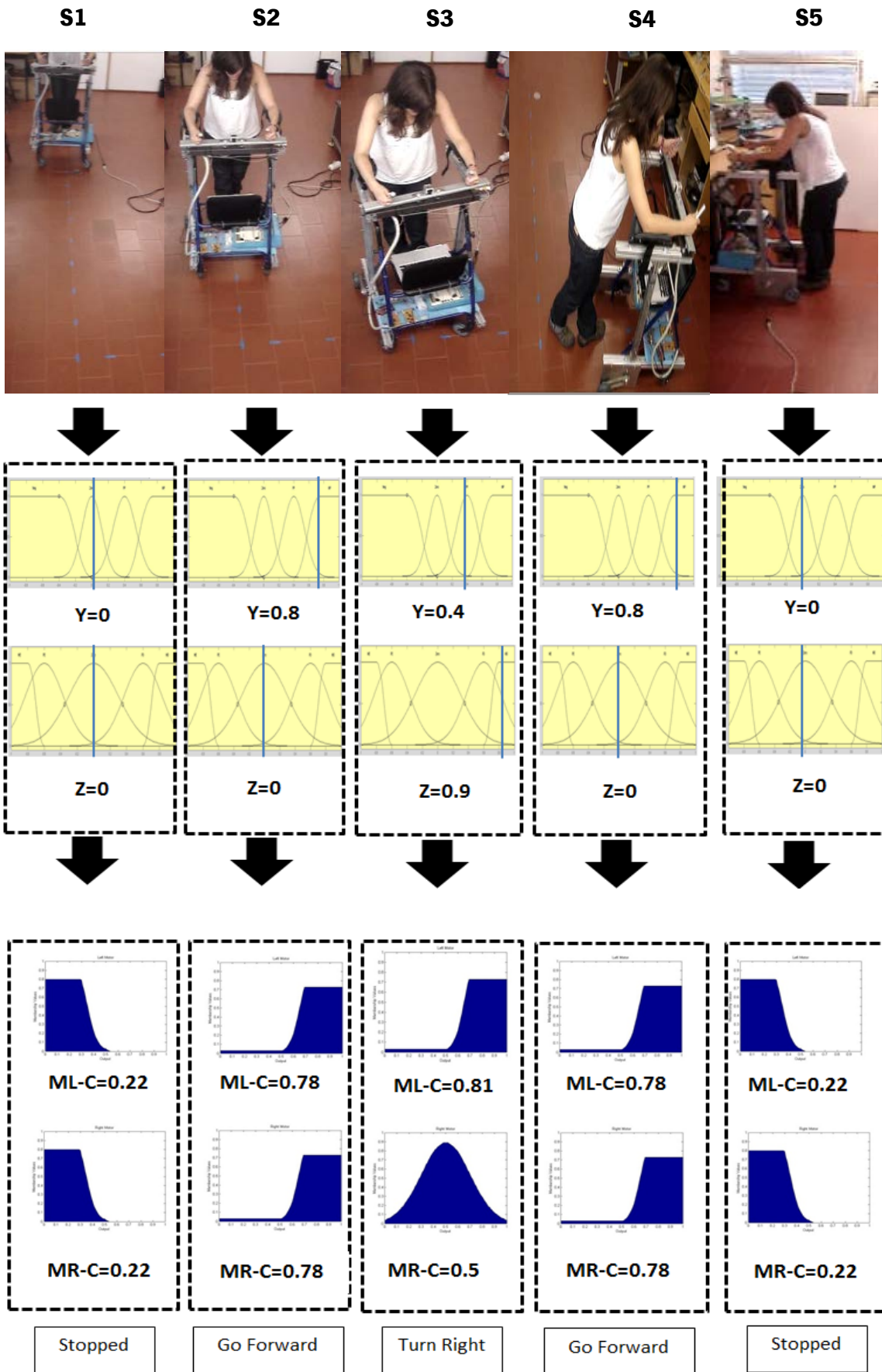


Figure 6.10 Scheme of the behaviour of the fuzzy system while the user is performing trajectory A.

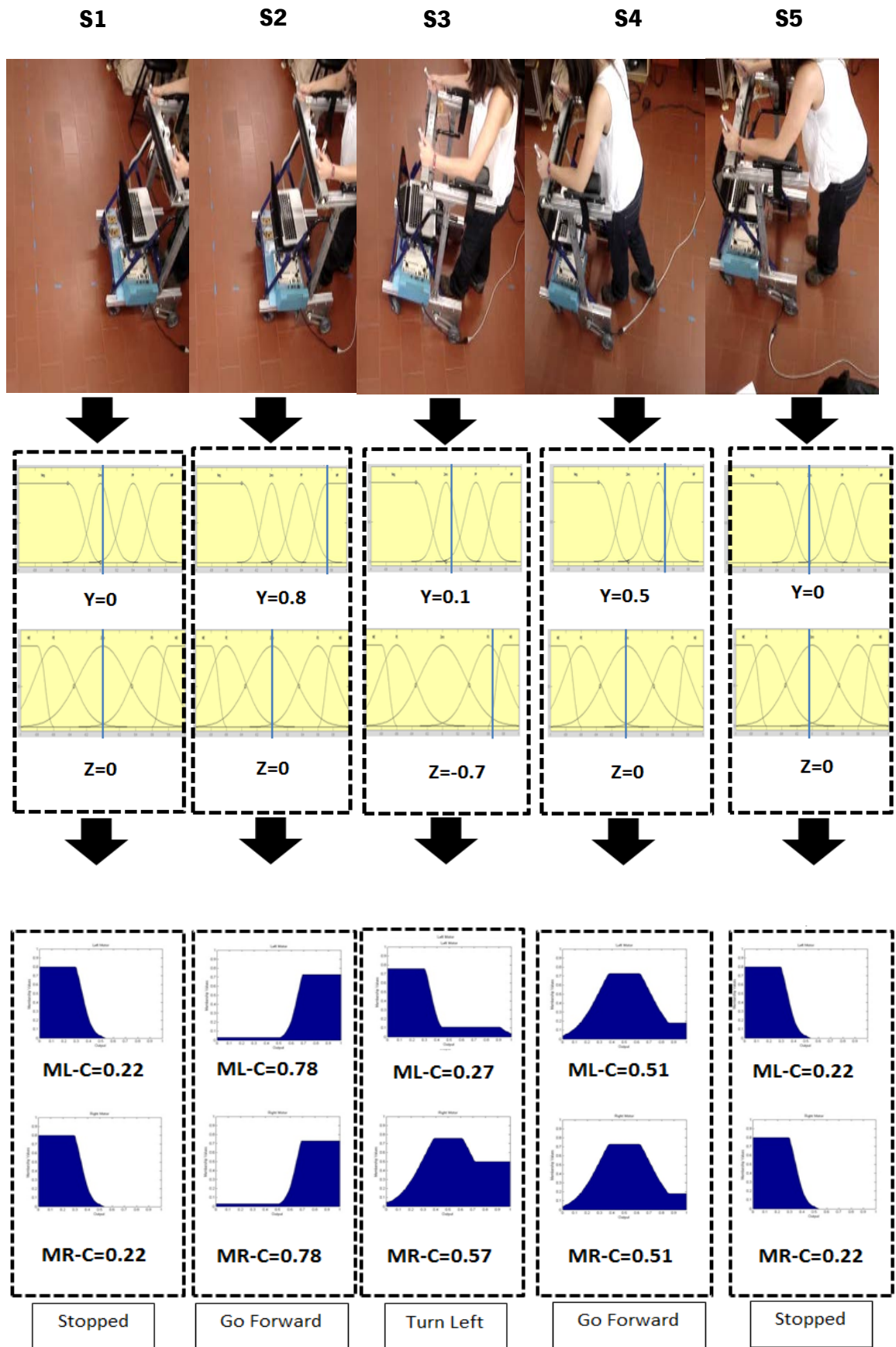


Figure 6.11 Scheme of the behaviour of the fuzzy system while the user is performing trajectory B.

6.3.3. Tuning, validation and results of the control strategy

A series of experiments with healthy users were conducted to assess the functioning of the fuzzy system and allow the tuning of the gains and parameters of the implemented system. The quality of the fuzzy system model depends on the proper tuning of the parameters of the MF

i) NeoASAS walker

The complete control system of the NeoASAS walker was designed in Simulink and then sent to the PC/104 platform through the MATLAB Real-Time Workshop.

The final designed control flow system after the adjustments of the gains and parameters of the system is shown in figure 6.12.

The “Compens offset” block eliminates the offset from the acquired signals and the resultant signals enter in the control block. In this block the signals are filtered by the BBF filter (BBF block), amplified according to gains, G and $G1$, restricted (ZM and Sat) and processed by the Fuzzy System. The outputs of the Fuzzy system are restricted (ZM), smoothed (integ block), amplified ($G2$ and $G3$) and converted to the range of $[2.5; 5]$ so that these could be sent to the control board hardware.

The BBF filter parameter (g) was maintained during the experiments with the walker. This block, as it was shown in chapter 4, is required to eliminate/attenuate the noise of the acquired joystick signals. The ZM block is the dead zone block, and it prevents that small values of signal amplitude close to zero correspond to voltages that would activate the motors when is not necessary. The SAT block is the saturation block and it prevents the signal to have a value above or below a certain range, that in this case is $[-1,1]$ for the input signals and $[0,1]$ for the output signals. The amplification blocks of the inputs are necessary to adjust the signals amplitude to the range of $[-1,1]$ to then enter in the fuzzy system block and to improve the sensibility of the system to the signals. The integ block serves to smooth the output signal.

While walking with the walker, the users felt that the movement of the wheel was not very smooth, and the resulting driving of the walker was instable. So after the fuzzy control, it was added an integrator block to smooth the signal, preventing the system from becoming unstable. This will cause the guidance of the walker to be smoother. However, it is necessary to take into

account that the integrator can cause the delay of the signal, causing the walker's responses to be slower. It is, therefore, necessary to take into consideration the trade-off between the signal response and its smoothness.

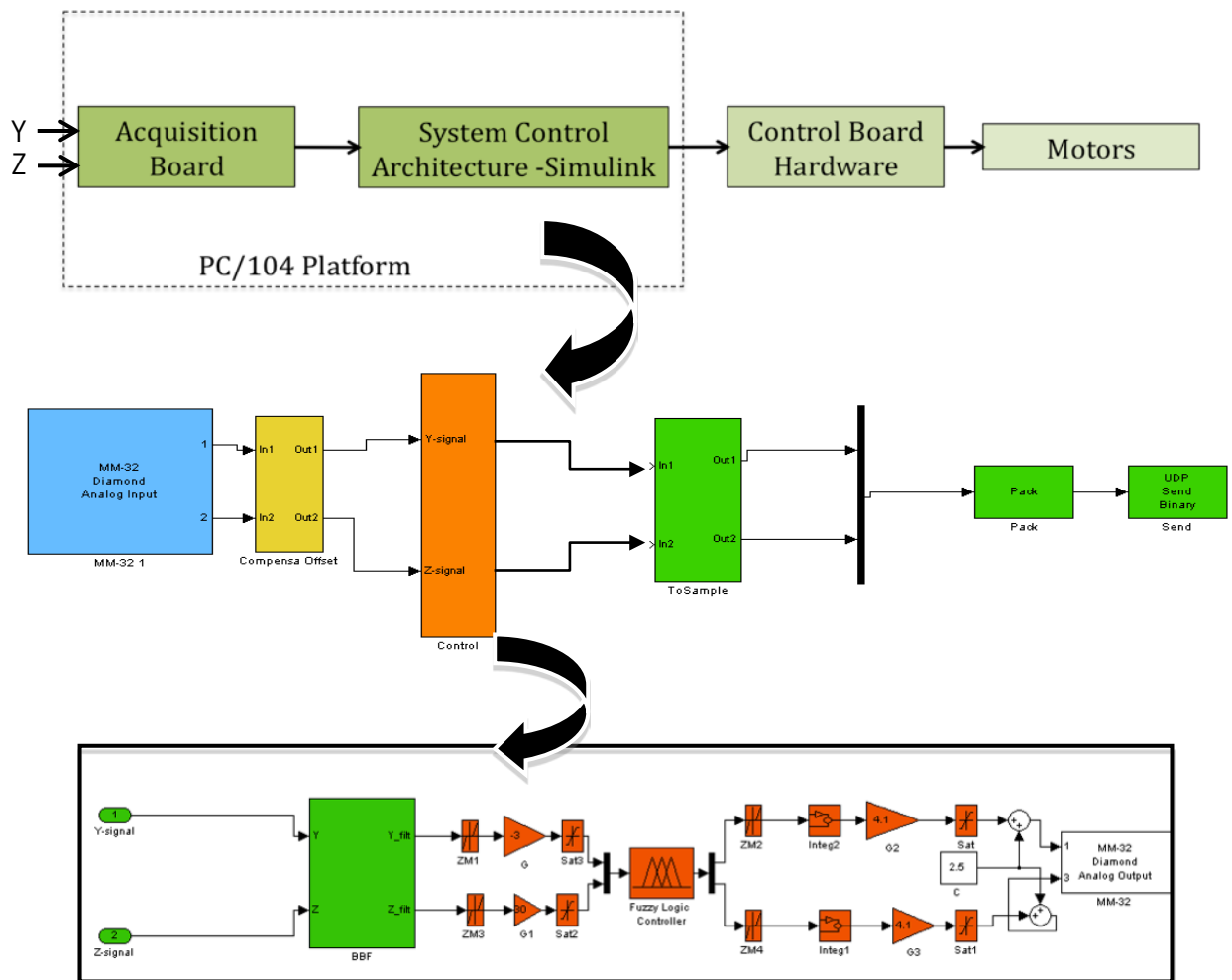


Figure 6.12 System Architecture and control flow of the NeoASAS walker designed in Simulink. The control block is presented in detail.

An example of the results obtained using the NeoASAS walker control system is presented in figures 6.13 and 6.14.

In figures 6.13 and 6.14, two examples of results are shown. The signals were acquired while a user was performing the trajectory A and B (Chapter 4), respectively. Looking at panel a) in both figures it can be seen that the presence of the motors in the movement of the walker, led to a Y and Z-signal with a more accentuated noise, comparing with the one saw in chapter 4. This is because of the increase of the vibrations and electromagnetic noise due to the motors. However,

as it is shown in these same figures, the result from the filter BBF is very satisfactory, proving the good performance of the filter in attenuating the noise components.

So, in what concerns the BBF filter, the real implementation did not required any adjustments, thus the values were kept as they were presented in chapter 4.

The Y and Z-signals represented here are similar to the one's showed in chapter 4, in the way that present the same response for the trajectory A.

In panel b) it is shown the result obtained after the BBF filter block, as well as the gain, ZM and SAT blocks. The Y-signal was inverted, amplified 3 times and it is in the range of $[-1,1]$. The Z-signal is also in the range of $[-1,1]$, it was amplified 30 times, and here the actuation of the ZM block can be noticed, as the little variations near zero were not amplified. These results can be verified if one compare these figures with the ones showed in a).

In panel c) one can see the output of the fuzzy control system as well as the result from the integ block. Note that the signal was converted to the range of $[2.5,5]$ to be sent to the control board hardware to be finally sent to the motors.

A more descriptive analysis follows. The most important instants of time, described in the text, are identified in these figures by thick black vertical lines.

In figure 6.13, the user performs trajectory A. At $t=1s$, the user transmits to the walker the intention to start walking. This can be seen in figure 6.13b, where the Y-signal increases its amplitude denoting an intention of the user to go forward. Consequently, both motors start to increase their velocity, as it can be seen in figure 6.13c at $t\approx 1s$.

At $t\approx 9s$, the user wants to turn right, as it is shown in b), and the Z-signal turns positive (Chapter 4). This intention is reflected in the motors movement where the right motor decreases its velocity, and the left one maintains its. The walker starts to turn right, as the user commanded. At $t\approx 13s$ the curve ends.

At $t\approx 16s$, the user's intention is to stop, so he pushes the handles to himself, decreasing the amplitude of the Y-signal. At the same time, the Z-signal also presents an increase in its amplitude. This happened because the user pushed the handles asymmetrically. However, this will not influence the movement of the walker, since it was implemented the rule that the walker cannot move backwards. As expected, one can verify in figure 6.13c that at $t\approx 16s$ the motors stopped.

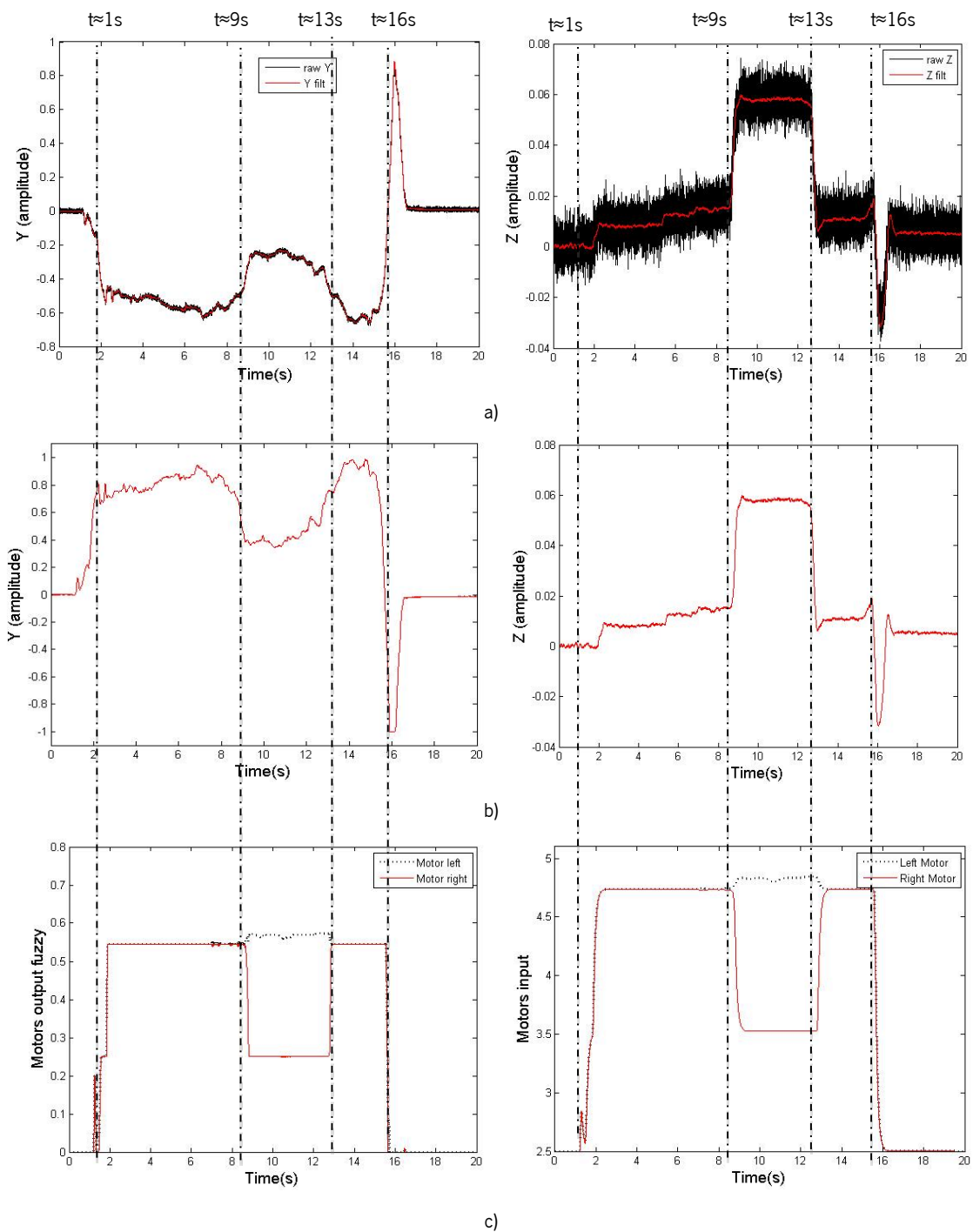


Figure 6.13 Results from the system architecture of the NeoASAS walker acquired while the user was performing the trajectory A. a) raw acquired joystick signals and the result from the filtering with the BBF filter; b) signals before the amplification and restrictions; c) output of the fuzzy system (left) and the conversion and integration of these results to then be sent to the control board hardware (right).

Looking now to figure 6.14 where the user performs the trajectory B, one can see that the user transmits to the walker at $t \approx 2s$ the intention to start walking. This can be seen in figure 6.14b, where the Y-signal increases their amplitude, informing the walker the intentions of the user to go

forward. Consequently, both motors start to increase their velocity, as it can be seen in figure 6.14c at $t \approx 2$ s. At $t \approx 4$ s, the user decided to accelerate more. This was concluded by the observation of figure 6.14b, where the Y-signal amplitude increases even more, as well as the motors signals (Figure 6.14c).

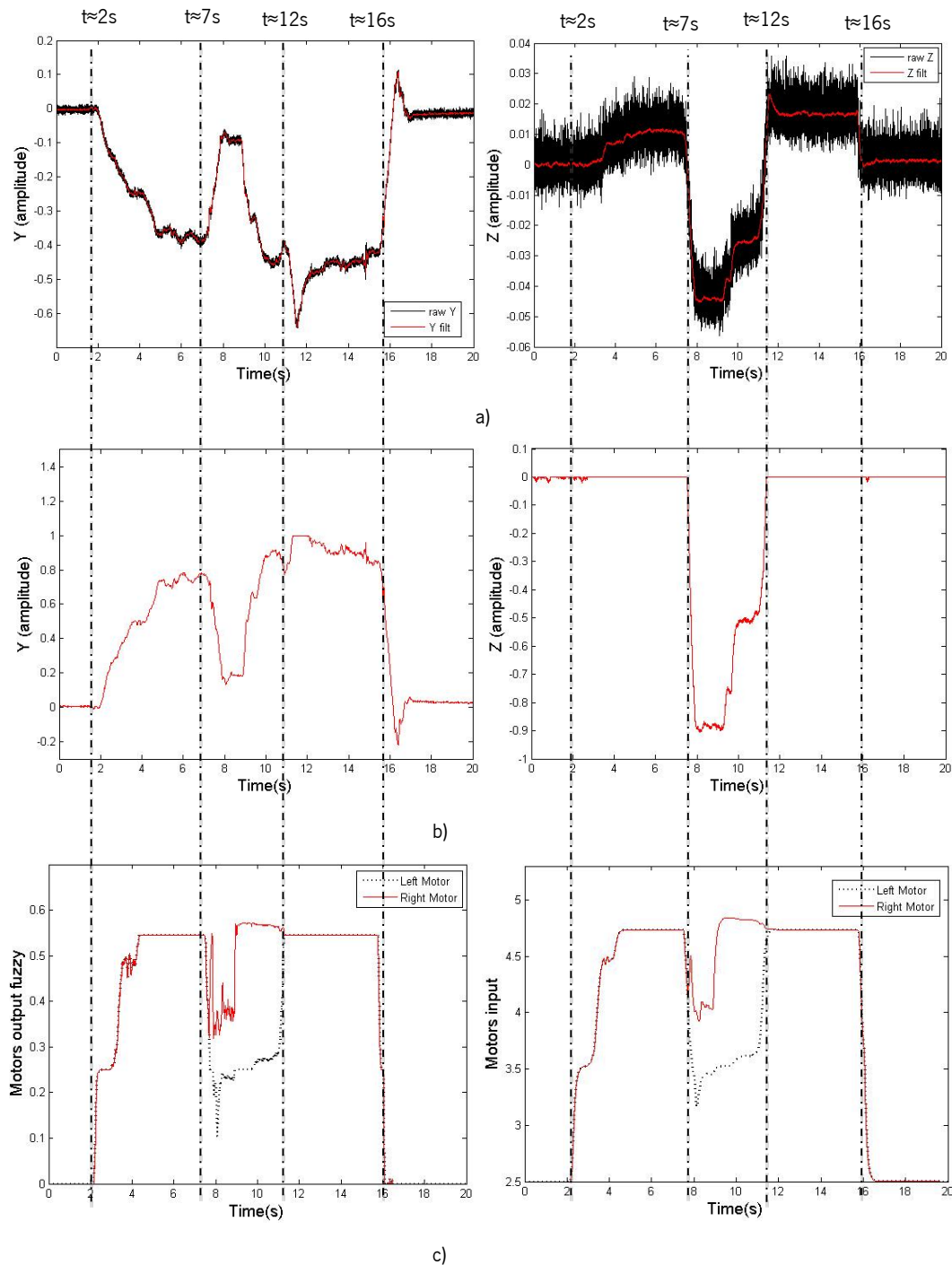


Figure 6.14 Results from the system architecture of the NeoASAS walker acquired while the user was performing the trajectory B. a) raw acquired joystick signals and the result from the filtering with the BBF filter; b) signals before the amplification and restrictions; c) output of the fuzzy system (left) and the conversion and integration of these results to then be sent to the control board hardware (right).

At $t \approx 7s$, the user wants to turn left, as it is shown in b), where the Z-signal turns negative (chapter 4). This intention is reflected in the motors movement where the left motor decreases its velocity, and the right one maintains its. This way, the walker starts to turn left, as the user commanded. Then, the user transmits to the walker that he wants to increase the velocity of the execution of the curve, passing of a translation curve to a rotational curve. This happens at $t \approx 10s$, where the amplitude of Y-signal is increased. Thus, the right motor increases its velocity, as it can be seen in figure 6.14c. The intention to turn left ends at $t \approx 12s$.

At $t \approx 16s$, the user's intention is to stop, so he pushes the handles to himself, decreasing the amplitude of the Y-signal. This is shown in figure 6.14c where at $t \approx 16s$ the motors' signals is zero, which means that the motors stopped.

Despite the variations of the Y and Z-signal, the motors present a constant and safe movement. Therefore, the system is perfectly adjusted to read the user's command intentions.

Thus, it was successfully generated a control strategy which has low computational cost, allowing a smooth and enjoyable driving, fast response of the walker and no sense of delay. The user feels that has the control of the device and that it follows his intentions to move.

i) ASBGo walker

The complete control system of the ASBGo walker is shown in figure 6.15.

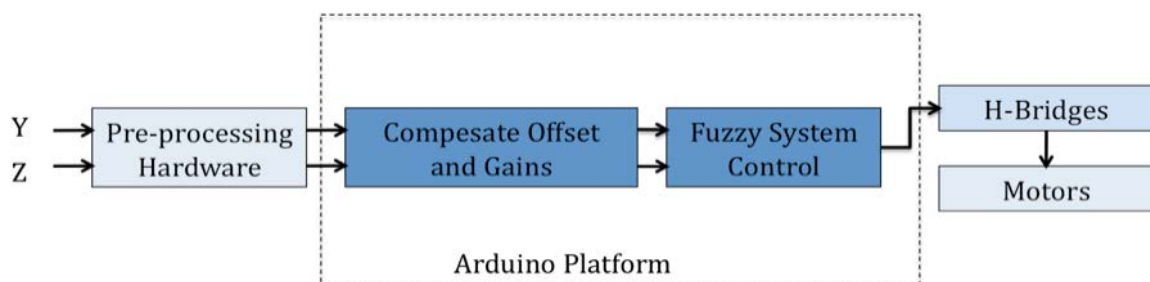


Figure 6.15 A schematic of the complete control architecture system used in the ASBGo walker. It starts with a circuit hardware block that sends signals to the Arduino platform. Then the outputs are sent to the H-bridges, which power the motors.

The raw joystick signals are first processed in the Pre-processing Hardware block. This is compounded by two parts, filtering and amplification, as described in Chapter 4. Then the joystick signals are sent to the Arduino platform where the offset of the Y-signal (if it exists) is eliminated as well as the offset of 1.6V of the Z-signal. After this, these signals are converted to the range of $[-1,1]$ and some restrictions are made for the amplitudes of the two signals, according to a dead

zone range (ZM) as well as a saturation limits (SAT). Finally, the signals are processed in the Fuzzy System Control Block, where they are also smoothed and then sent as PWM outputs from the Arduino platform to the H-bridges. These components convert these digital signals into higher power signals to then power the motors.

In figures 6.16 and 6.17 the results from these control systems when a user was performing trajectories A and B, respectively, are shown.

Panel a) shows the resultant signals from the pre-processing hardware block. These signals are similar to the ones showed in Chapter 4. Despite the increase of the vibrations from the structure, this block still performed a good signal filtering.

Panel b) depicts the signals after the elimination of the offset, conversion into the range of $[-1,1]$ and restrictions. The dead zone restriction is more noticed in the Z-signal, in which small variations around zero were eliminated.

Panel c) depicts the results from the fuzzy control system, as well as the ones after being smoothed. Then these outputs are sent as PWM outputs to the H-Bridges block of the control system.

A more formal analysis of these figures follows.

In figure 6.16 the user performs trajectory A with the ASBGo walker.

In figure 6.16 one can see through the Y-signal that the user begins to walk at $t \approx 0.5s$ and consequently both motors acquire the same velocity to start moving forward. Then, the user accelerates until $t \approx 2s$, after which he kept his velocity. Despite the variations of the Y-signal, the resultant signal sent to the motors is stable. At $t \approx 5.5s$, the user begins to turn right until $t \approx 7.5s$ and the Z-signal reflects this intention. This command is transmitted to the motors, and the left motor loses velocity. At $t \approx 9s$, the user stops and the Y-signal becomes zero, as well as the signal send to the motors.

At $t \approx 9s$, the user stops since the Y-signal turns zero, as well as the motors signal.

Figure 6.17 depicts the acquired results while a user was performing trajectory B. This figure is built similarly to figure 6.14 The user begins to walk at $t \approx 1.5s$ as shown by the Y-signal (Figure 6.17a). Figure 6.17c shows the motors starting approximately at the same time to increase their velocity to move forward. Then, at $t \approx 6s$, the user turns left (Z-signal), ends the curve at $t \approx 9s$ and at $t \approx 11s$, the user stops, and the Y-signals decrease to zero, as well as the motors signals.

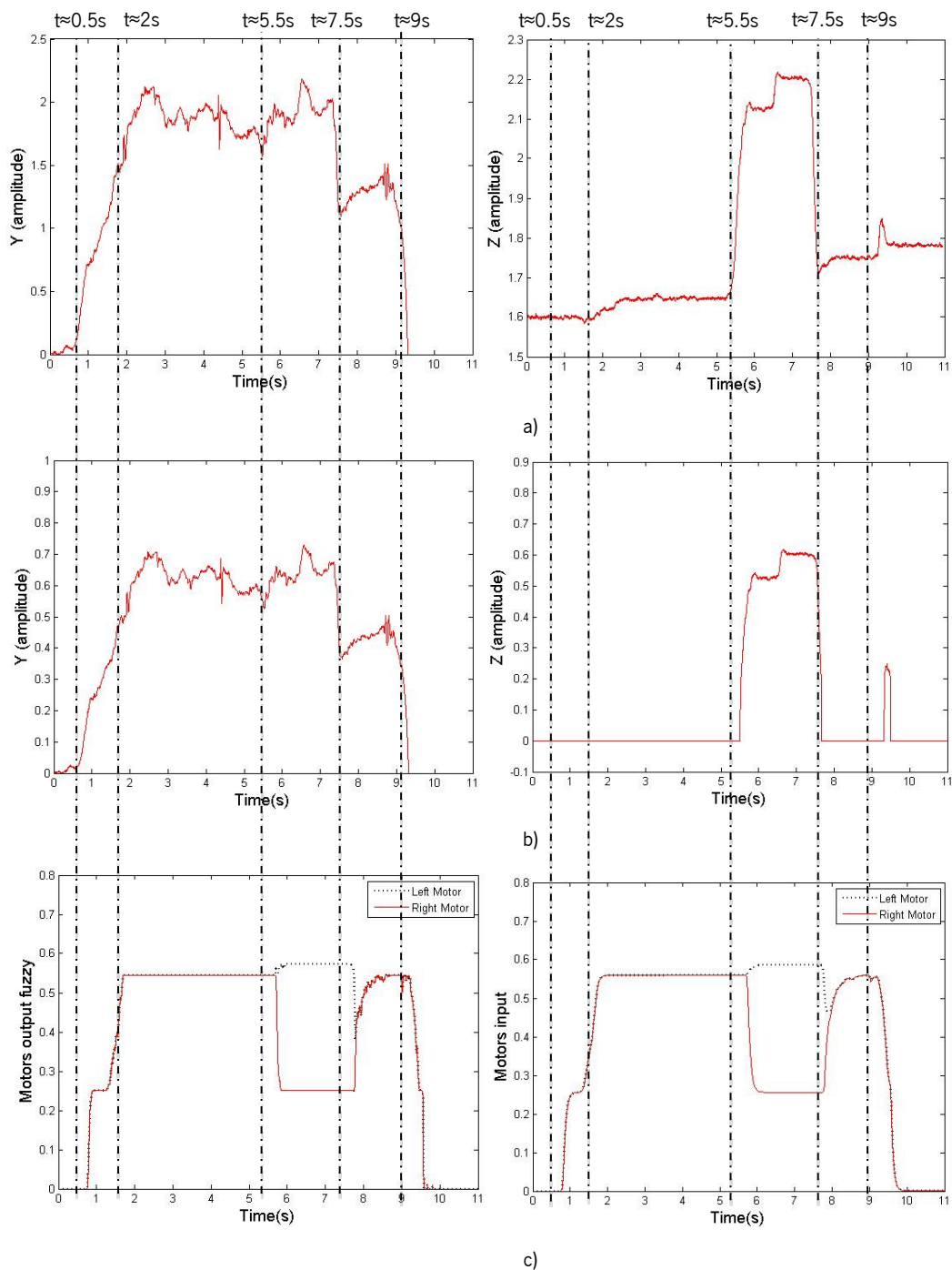


Figure 6.16 Results from the ASBGo system of the acquired while the user was performing trajectory A. a) acquired joystick signals after the pre-processing with the hardware circuit. b) signals before the amplification and restrictions in the Arduino platform; c) output of the fuzzy system (left) and then the conversion and integration of these results to then be sent to the H-bridges and then to the motors (right).

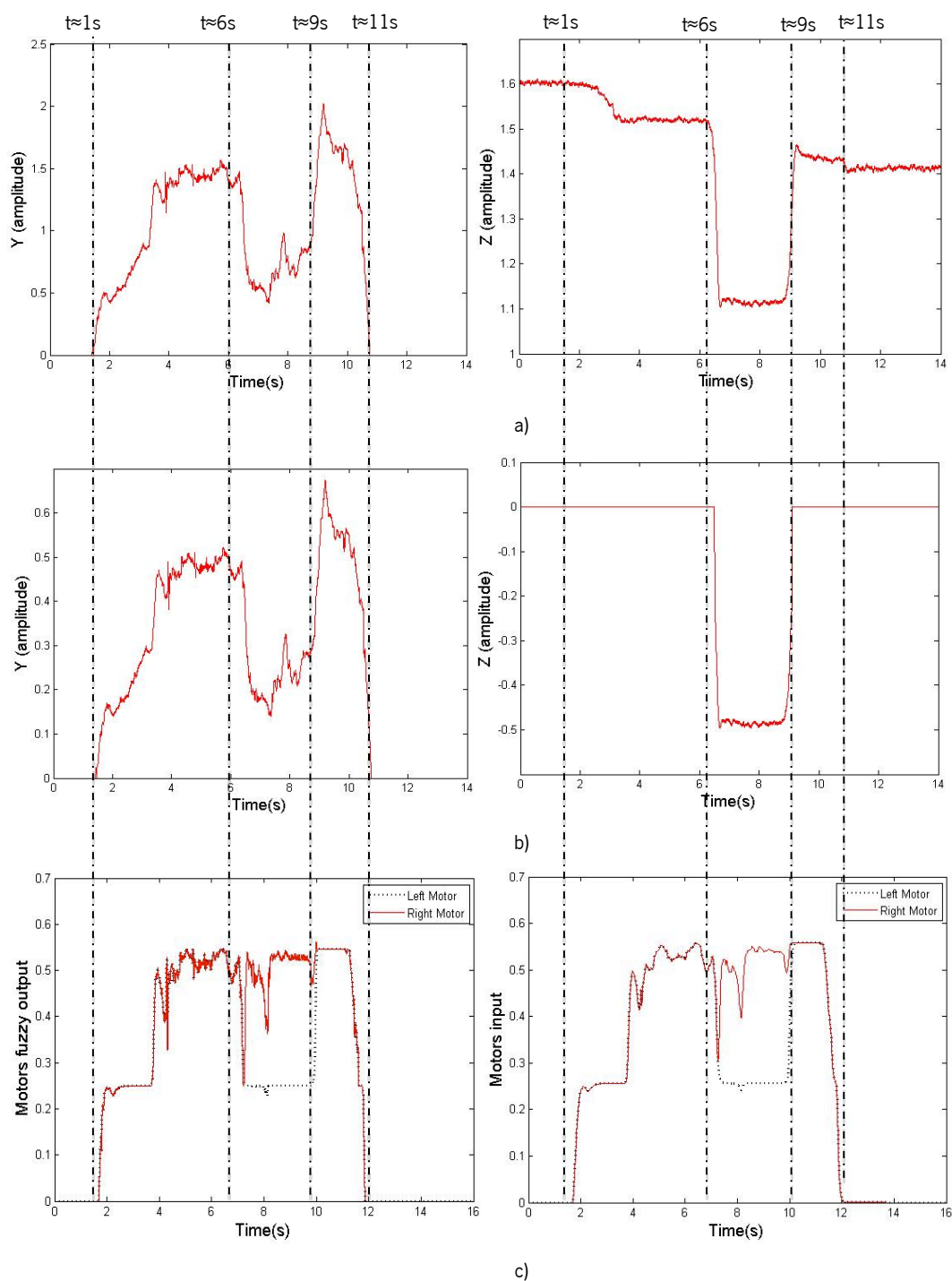


Figure 6.17 Results from the ASBGo system architecture acquired while the user was performing trajectory B. a) acquired joystick signals after the pre-processing with the hardware circuit . b) signals before the amplification and restrictions in the Arduino platform; c) output of the fuzzy system (left) and then the conversion and integration of these results to then be sent to the H-bridges and then to the motors (right).

Observing both figures 6.16c and 6.17c and comparing with the respective a), it is possible to verify a little delay on the response of the motors. Users complained to feel a delay between their commands and the motors response. This delay is around the 200 ms, and as it was seen before, 200 ms is the time of human perception. Thus this constitutes a drawback in the system platform that needs to be improved. This will be achieved by optimizing the employed algorithm and changing the control platform or the level of programming.

6.4. Conclusion

A control strategy that allows the walker to be driven intelligently through the use of a user-friendly interface was presented in this Chapter.

This control strategy is based in a fuzzy inference system, and it classifies the signals sent by the joystick, transforming them into motor outputs. This algorithm acts like a *smart* interpreter of the user's command intentions sent through the joystick that gives orders to the motors to follow the user's intentions.

The NeoASAS control system architecture using the PC/104 showed very good results, generating a control strategy with low computational cost, allowing a smooth and enjoyable driving, fast response of the walker and no sense of delay. The user feels that has the control of the device and that it follows his intentions to move.

However, the PC/104 is not a low cost platform, which makes it a drawback to the commercial focus of the device.

In the case of the ASBGo control system architecture, the results were good in terms of acquisition, filtering, amplification and fuzzy control response. However, the time that the system takes to process the entire algorithm is very significant, *i.e.* the user has the sense of delay while using the walker.

So, despite being a low cost and simple approach, some improvements have to be made to increase the performance of this platform. However, these considerations were not taken into account during the development of the proposed work and more care was given to the overall development of the walker and its assemblage. These improvements will be tackled in a near future.

7. Conclusions and Future Work

7.1. Conclusions

This section presents the conclusions of this work.

In Chapter 2, it was presented a review of the state of the art in the robotic technology for mobility assistive devices for people with mobility disabilities. The important role that robotics can play in mobility assistive devices was presented, as well as the identification and survey of mobility assistive devices subsystems with a particular focus on the walkers technology. The advances in the walkers' field have been enormous and have shown a great potential on helping people with mobility disabilities. Thus, it was presented a more focused state of the art of walkers, more specifically Smart Walkers, and it was discussed major advances that have been made and limitations to be overcome.

These studies enabled to conclude that there are important flaws in the Smart Walkers, considering security, stability, physical support, cost and, most important, the direct interaction with the user. Hence, there is effectively space and necessity of an economic Smart Walker that provides for proper support and security, being centred on the user and presenting a user-friendly interface that helps the user with the driving of the walker.

In Chapter 3, it was done a brief presentation of the ASAS and SIMBIOSIS projects, followed by a presentation of the goals of the walkers covered in this thesis. Therefore, the NEOASAS and ASBGo projects emerged in order to give continuity to the ASAS and SIMBIOSIS projects, but with a commercial focus.

The NeoASAS and ASBGo aim to provide for good support and stability. These projects also aim try to offer a certain degree of intelligence in assistance and judgement through the integration of several sensors that inform the Smart Walker about the status of the user, *i.e.* identify the movement intentions of the user considering direction and velocity through a joystick placed onto the upper base of the walkers, as well as additional safety with infrared and force sensors, by detecting possible falls of the user.

Therefore, the interface is based on a joystick and it is intended to be user-friendly, simple, efficient, low cost and with little electronics.

Chapter 4 presented a series of experiments performed by healthy users, which showed the

sensibility of the joystick to extract intention commands from the user.

These volunteers had to perform simple tasks like moving forward and then turn left or right. It is noteworthy that these tests were performed without any motorized system to enable the real interaction evaluation between the user and the upper base support of the walker without the interference of any control strategy.

The joystick outputs provided two types of signals: Y-signal representing the straight or forward movement intention of the user and Z-signal representing the curvilinear movement intention of the user.

The analysis of the acquired data allowed to identify two main components in both Y and Z-signals. The first one represents the highest frequency noise caused by the vibrations of the structure due to the irregularities of the ground, electromagnetic noise and wheel eccentricities. The second component represents the voluntary component that contains the information of the walking movement intentions of the user to guide the walker.

Each component has been properly studied and processed. The first component had to be eliminated in real-time, not causing a considerable delay on the signal. Different approaches were taken for the two walkers. In the NeoASAS walker, this component was attenuated by a Benedict-Bordner $g-h$ filter. The g parameter was chosen to be 44.29×10^{-3} for the Y-signal and 16.87×10^{-3} for the Z-signal. The chosen filter presented low computational costs, making it suitable to this application, since it can run in a low cost hardware with enough robustness for a commercial device.

In the ASBGo walker, a proposed hardware circuit was mounted, and a pre-processing of the joystick's signals was done. This circuit included a filter that showed a good performance on attenuating the vibration component. The performed different experiments allowed to conclude that a digital filter strategy was not necessary, allowing the reduction of the computational cost of the controller algorithm.

The second component contains the voluntary movement information that can be used to guide the walker. It enables to characterize the trajectory that the user pretends to follow, making possible the continuous monitoring of the users' movement, allowing the development of robust and secure control strategies.

In Chapter 5 were presented the different safety mechanisms considered in the developed walker.

To identify if the user is falling forward, an IR sensor was integrated in the upper base of the walker structure. This sensor measures the distance between the user and the walker, successfully, and with the help of a developed algorithm, the walker is advised to stop when the distance between the user and the walker decreases significantly.

To identify if the user is falling backwards, *Flexiforce* sensors were integrated in the forearms support as well as in the handles of the upper base of the walker structure. This sensor measures the force that is made by the user when he relies onto the supports; and the force that he makes when grasping the handles. A proposed algorithm advises the walker to stop when no force is applied onto the support and/or in each handle.

Finally in chapter 6 it was built a control strategy based in a fuzzy inference system that classifies the filtered signals, transforming them into motor outputs. This works like a *smart* interpreter of the user's command intentions sent through the joystick that gives orders to the motors to follow the user's intentions. The user feels that has the control of the device and that it follows his intentions to move.

The NeoASAS control system architecture using the PC/104 showed very good results, allowing a smooth and enjoyable driving, fast response of the walker and no sense of delay. The user felt that he was in control of the device. However, the PC/104 is not a low cost platform, which makes it a drawback to the commercial focus of the device. Thus, another platform has to be built toward a commercial application.

The ASBGo control system architecture also showed good results. However, the time that the system takes to process the entire algorithm is very significant, *i.e.* the user has the sense of delay while using the walker.

So, despite being a low cost approach, some improvements have to be made to increase the performance of this platform. These improvements will be tackled in a nearby future.

7.2. Future Work

In future work a necessary step is the improvement of a Smart Walker capable of supporting the weight of the user but that can be used to stably and safely drive him with a high manoeuvrability. For instance, different handle bar designs must be addressed and it must be verified the best way to dispose the base support of the upper limbs of the user, in order to improve the manoeuvrability

of the walker and to improve the ergonomics. The walker will be based on the ASBGo project and also on the knowledge acquired while working with it.

Additionally, it is also necessary to find components that are low consumption and the battery system has to be improved. For safety the electrical system has to be isolated from the environment in an enclosure to prevent damage from liquid-spills, etc. The re-charge time of the batteries should not be longer than eight hours to achieve full charge thus allowing re-charge overnight. And the runtime of the system should be of the order of six hours. The re-charge system should be simple to connect and monitor.

The results obtained with these modifications should be used to advance towards a commercial product with an affordable cost but high reliability and safety. The motivation is that this will contribute towards better rehabilitation purposes by promoting ambulatory daily exercises and thus extend users' independent living and improve their life quality.

The filter and control strategies should be adaptive and dynamic. In this work a static approach showed very good results, however with healthy users. These people are more stable and predictive than the pathological users that may present unpredictable behaviours, and sometimes have asymmetrical gait patterns, etc. Thus, the objective is to provide assisted locomotion adapted to the user's needs and to short- and long-term changes in mobility capacities and modifications on gait patterns.

To validate this it must be promoted contacts with both medical staff and clinics for medical evaluation and quantitative assessment of the walker's potential as a rehabilitation and functional compensation too.

References

- [1] World Population Ageing 1950-2050. Population Division, DESA, United Nations.[<http://www.un.org/esa/population/publications/worldageing19502050/pdf/81chapteriii.pdf>.] Accessed in 25 May 2010.
- [2] Ceres R :Las tecnologías de apoyo para la vida independiente de los mayores. Entre MayoresEM 2007, 10-11.
- [3] Raya R, Frizera A, Pastor J, Ceres R, Calderón L: Configuración de un sistema ultrasónico multisensor para navegación asistida de niños con deficiencias neuromotoras. Actas de las XXVIII Jornadas de Automática, 2007.
- [4] Ceres R, Pons J L, Calderón L, Jiménez AR, Azevedo L: A robotic vehicle for disabled children. IEEE Engineering in Medicine and Biology Magazine 2005, 24(6):55–63.
- [5] Lacey G, Dawson-Howe K: Evaluation of Robot Mobility Aid for the Elderly Blind. Proceedings of the Fifth International Symposium on Intelligent Robotic Systems, 1997.
- [6] Duxbury A: Gait Disorders and Fall Risk: Detection and Prevention. Comp Ther, 2000, 26(4):238–245.
- [7] Axer H, Axer M, Sauer H, Witte O, Hagemann G: Falls and gait disorders in geriatric neurology. Clinical Neurology and Neurosurgery 2010, 112, 265–274.
- [8] Lundin-Olsson L, Nyberg L, Gustafson Y: Stops walking when talking as a predictor of falls in elderly people. Lancet 1997, 349: 617.
- [9] Anke H, Snijders P, van de Warrenburg N, Bastiaan R, Bloem: Neurological gait disorders in elderly people: clinical approach and classification. Lancet Neurol 2007, 6(1):63-74.
- [10] Constantinescu R, Leonard C, Deeley C, Kurlan R. : Assistive devices for gait in Parkinson's disease. Parkinsonism and Related Disorders 2007, 13:133–138.
- [11] Pires G, Rui A, Urbano N, Almeida A: Robchair: A Powered Wheelchair using a Behaviour-Based Navigation. Proc. IEEE Int. Workshop on Advanced Motion Control (AMC'98) 1998, 536-541.
- [12] Reis L, Braga R, Sousa M, Moreira A : IntellWheels MMI: A Flexible Interface for an Intelligent Wheelchair. Lecture Notes in Computer Science 2010, 5949/2010:296-307, DOI: 10.1007/978-3-642-11876-0_26.
- [13] Nisbet PD, Craig J, Odor JP, Aitken, S: “Smart” wheelchairs for mobility training. Technology and Disability 2005, 49–62.

- [14] Image wheelchair [http://www.wheelchairslift.net/wheelchair/manual-wheelchairs-3/]. Accessed in 29 May 2010.
- [15] Tanaka K, Matsunaga K, Wang HO: Electroencephalogram-based control of an electric wheelchair. *IEEE Transactions on Robotics and Automation* 2005, 21:762–766.
- [16] Frizzera A, Celeste WC, Martins VR, Bastos-Filho TF, Sarcinelli-Filho M: Human-machine interface based on electro-biological signals for mobile vehicles. In *Proceedings of the International Symposium on Industrial Electronics* 2006.
- [17] Ferreira A, Silva RL, Celeste WC, Bastos Filho TF, Sarcinelli Filho M: Human-machine interface based on muscular and brain signals applied to a robotic wheelchair. *Journal of Physics* 2007. DOI: 10.1088/1742-6596/90/1/012094.
- [18] Andrzej S , Cichocki, Riken-Toyota Collaboration Lab Head. *Journal Article* 2009. [http://cacm.acm.org/news/43117-thought-controlled-wheelchair/fulltext]. Accessed in 19 May 2010.
- [19] Exmovere.[http://www.exmovere.com/mobility.html] Accessed in 19 May 2010.
- [20] Medical Supplies & Equipment Company. [http://rehabilitation-products.medical-supplies-equipment.com/PPF/page_ID/195/article.asp]. Accessed in 29 May 2011.
- [21] Novandy B, Yoon J, Manurung A: Interaction Control of a programmable Footpad-Type Gait Rehabilitation Robot for Active Walking on Various Terrains. *IEEE 11th International Conference on Rehabilitation Robotics Kyoto International Conference Center* 2009, 23-26.
- [22] Colombo G: Treadmill training of paraplegic patients using a robotic orthosis. *Journal of Rehabilitation Research and Development* 2000, 37 (6): 693-700.
- [23] Lueningburger L, Colombo G, Riener R: Biofeedback for robotic gait rehabilitation. *Journal of NeuroEngineering and Rehabilitation* 2007, 4:1.
- [24] Chin LF, Lim WS, Kong KH: Evaluation of robotic-assisted locomotor training outcomes at a rehabilitation centre in Singapore. *Singapore Med J* 2010 51(9): 709-715.
- [25] Schwartz I, Sajin A, Fisher I, Neeb M, Shochina M, Katz-Leurer M, Meiner Z: The Effectiveness of Locomotor Therapy Using Robotic-Assisted Gait Training in Subacute Stroke Patients: A Randomized Controlled Trial. *Medical Association Journal* 2009, 1:516-523.
- [26] Swinnen E, Duerinck S, Baeyens J, Meeusen R, Kerckhofs E: Effectiveness of robot-assisted gait training in persons with spinal cord injury: a systematic review. *Journal of Rehabilitation Medicine* 2010, 42:1-7.
- [27] Veneman J, et al.: Design and Evaluation of the LOPES Exoskeleton Robot for Interactive Gait

Rehabilitation. IEEE Transactions on Neural Systems and Rehabilitation Engineering 2007, 15 (3):379-386.

[28] Leahy, Theresa E. Impact of a limited trial of walking training using body weight support and a treadmill on the gait characteristics of an individual with chronic, incomplete spinal cord injury. *Physiotherapy theory and practice* 2010, 26(7):483-489.

[29] Pillar, Thomas, Dickstein R, Smolinski, Z: Walking Reeducation with Partial Relief of Body Weight in Rehabilitation of Patients with Locomotor Disabilities. *Journal of Rehabilitation Research and Development* 1991, 28(4):47-52.

[30] Veg A, Popovic D: Walkaround: Mobile Balance Support for Therapy of Walking. IEEE Transactions on Neural Systems and Rehabilitation Engineering 2008, 16(3).

[31] Bourri M, Stauffer Y, Schmitt C, Allemand Y, Gnemmi S, Clavel R, Metrailler P, Brodard R: The WalkTrainer: A Robotic System for Walking Rehabilitation. IEEE International Conference on Robotics and Biomimetics. ROBIO '06.

[32] The Rehabilitation Institute of Chicago. (2005). [<http://www.kineadesign.com/portfolio/kineassist/>]. Accessed in 19 May 2011.

[33] Seo K, Lee J: The development of two mobile gait rehabilitation systems. IEEE Transactions on Neural Systems and Rehabilitation Engineering 2009,17(2):156–166.

[34] LiteGait [<http://litegait.com/research.html>]. Accessed in 30 May 2011.

[35] Schmidt H, Werner C, Bernhardt R, Hessen S, Kruger J: Gait rehabilitation machines based on programmable footplates. *Journal of NeuroEngineering and Rehabilitation* 2007.

[36] Monaco, V., Jung, J. H., Macri, L., Bagnato, S., Micera, S., Carrozza, J. H., Galardi, G.. Robotic system for gait rehabilitations of stroke patients during the acute phase. IEEE International Conference on Rehabilitation Robotics 2009.

[37] Nudo RJ, Milliken GW: Reorganization of movement representation in primary motor cortex following focal ischemia infarcts in adult squirrel monkeys. *Journal of Neurophysiology* 1996, 75:2144-2149.

[38] Dollar A, Herr H: Lower Extremity Exoskeletons and Active Orthoses: Challenges and State-of-the-Art. IEEE Transactions on Robotics 2008, 24(1).

[39] Pons, J. *Wearable Robots: Biomechatronic Exoskeletons*. John Wiley and Sons 2008.

[40] Kawamoto H, Lee S, Kanbe S, Sankai Y: Power assist method for HAL-3 using EMG-based feedback controller. In Proc. IEEE Int. Conf. Syst., Man, Cybern. 2003, 1648–1653.

[41] ARGO, Medical Technologies Ltd. [<http://www.argomedtec.com/products.asp>]. Accessed in 19

June 2010.

[42] Pratt JE, Krupp BT, Morse CJ, Collins SH: The RoboKnee: An exoskeleton for enhancing strength and endurance during walking. In Proc. IEEE Int. Conf. Robot. Autom. 2004, 2430–2435.

[43] Ferris D, Czerniecki J, Hannaford B: An ankle-foot orthosis powered by artificial pneumatic muscles. Journal of Applied Biomechanics 2005, 21 (2):189-197.

[44] Blaya JA, Herr H: Control of a variable-impedance ankle-foot orthosis to assist drop-foot gait. IEEE Trans Neural Syst Rehabil Eng 2004, 12(1):24-31.

[45] Moreno J, Rocon E, Ruiz AF, Brunetti F, Pons JL: Design and implementation of an inertial measurement unit for control of artificial limbs: application on leg orthoses. Sensors and Actuators B 2006, 118:333–337.

[46] Banala S, Agrawal S, Fattah A, Krishnamoorthy V, Hsu W, Scholz J, Rudolph K: Gravity-Balancing Leg Orthosis and Its Performance Evaluation. IEEE Transactions on robotics 2006, 22(6): 1128-1239.

[47] Hoo FW, Demonbreun D, Weiss BD: Ambulatory devices for chronic gait disorders in the elderly. American Family Physician 2003, 67(8):1717–1724.

[48] Spenko M, Yu H, Dubowsky S: Robotic personal aids for mobility and monitoring for the elderly. IEEE Transactions in Neural Systems and Rehabilitation Engineering 2006, 14(3):344–351.

[49] Borenstein J, Ulrich I: The guidecane - a computerized travel aid for the active guidance of blind pedestrians. In Proceedings of the IEEE International Conference on Robotics and Automation 1997, 1283 – 1288.

[50] Crutche: [<http://druffmix.com/soul-on-crutches/>] Accessed in 30 May 2010.

[51] Costa N, Caldwell D: Control of a biomimetic soft-actuated 10dof lower body exoskeleton. In Proceedings of the First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006, 495-501.

[52] Bateni H, Maki B: Assistive Devices for Balance and Mobility: Benefits, Demands, and Adverse Consequences. Arch Phys Med Rehabil 2005, 86.

[53] Molloy, D.. Boomer. [http://student.designawards.com.au/application_detail.jsp?status=6&applicationID=5541]. Accessed in 30 May 2010.

[54] Cub E, Moore CG, Leurgans S, Goetz CG: Wheeled and standard walkers in Parkinson's disease patients with gait freezing. Parkinsonism Relat Disord 2003, 10:9–14.

- [55] Standard Walker: [<http://www.technomedic.ca/>]. Accessed in 30 May 2010.
- [56] U-Step. [<http://www.ustep.com/walker.htm>]. Accessed in 30 May 2010.
- [57] Marincek C, Buhler C, Knops H, Andrich R: (2001) Assistive Technology: Added Value to the Quality of Life. Assistive Technology Research Series 2001.
- [58] Morris AC, Donamukkala RR, Kapuria A, Steinfeld A, Matthews J, Dunbar-Jacobs J, Thrun S: A robotic walker that provides guidance. In Proceedings of the 2003 IEEE Conference on Robotics and Automation 2003, 25–30.
- [59] Frizera A, Abellanas A, Ceres R, Pons JL, Raya R: Study and Characterization of Feet Kinematics in Walker Assisted Gait. *Revista Iberoamericana de Automatica e Informatica Industrial (RIAI)* 2009, 6(4): 74-84.
- [60] Tan R, Wang S, Jiang Y, Ishida K, Nagano M: Adaptive Controller for Motion Control of an Omni-directional Walker. Proceedings of the 2010 IEEE International Conference on Mechatronics and Automation. Xi'an, China, 156-161.
- [61] Frizera A, Ceres R, Pons J.L, Abellanas A, Raya R: The smart walkers as geriatric assistive device. The SIMBIOSIS purpose. *Gerontechnology (Valkenswaard)* 2008, 7:108-115.
- [62] Alwan M, Wasson G, Sheth P, Ledoux A, Huang C: Passive derivation of basic walker-assisted gait characteristics from measured forces and moments. In Proceedings of the 26th Annual International Conference of the IEEE EMBS 2004, 2691– 2694.
- [63] Ceres R, Pons JL, Calderón L, Mesonero-Romanos D, Jiménez VF, Sánchez X, Abizanda P, Saro B, Bonivardo G: Andador activo para la rehabilitación y el mantenimiento de la movilidad natural. *IMSERSO, Estudios I+D+I* 2005, 25:1 – 8.
- [64] Borgolte U: A novel mobility aid for independent daily living of elderly people. In Proceedings 5th European Conference for the Advancement of Assistive Technology (AAATE) 1999, 267 – 271.
- [65] Frizera A: Interfaz Multimodal para modelado, estudio y asistencia a la marcha mediante andadores robóticos –Phd Thesis, 2010, Universidad de Alcalá.
- [66] Tagawa Y, Shiba N, Matsuo S, Yamashita T: Analysis of human abnormal walking using a multi-body model joint models for abnormal walking and walking aids to reduce compensatory action. *Journal of Biomechanics* 2000, 33:1405–14.
- [67] Scherer MJ: The impact of assistive technology on the lives of people with disabilities. In: Gray DB, Quatrano LA, Lieberman ML, editors. *Designing and using assistive technology in human perspective*. Baltimore: PH Brooks, 1998, 99–116.
- [68] Mann WC, Granger C, Hurren D: An analysis of problems with walkers encountered by elderly

persons. *Phys Occup Ther Geriatr* 2005, 13:1–23.

[69] Charron PM, Kirby RL, MacLeod DA: Epidemiology of walker related injuries and deaths in the United States. *Am J Phys Med Rehab* 1995, 74:237–9.

[70] Lacey G, MacNamara S, Dawson-Howe KM: Personal adaptive mobility aid for the infirm and elderly blind. *Lecture Notes in Computer Science* 1998, 1458:211 – 220.

[71] MacNamara S, Lacey G: A smart walker for the frail visually impaired. In *Proceedings of IEEE International Conference on Robotics and Automatonomous* 2000.

[72] Cortés U, Martínez-Velasco A, Barrué C, Martín X, Campana F, Annicchiarico R, Caltagirone C: Towards an intelligent service to elders mobility using the i-Walker. *Proc. of AAAI Fall Symposia AI in Eldercare: New Solutions to Old Problems* 2008, Washington, USA.

[73] Wasson G, Gunderson J, Graves S: Effective shared control in cooperative mobility aids. In *Proceedings of the Fourteenth International Florida Artificial Intelligence Research Society Conference* 2001, 1:1-5.

[74] Dubowsky S, Genot F, Godding S, Kozono H, Skwersky A, Yu H, Yu LS: PAMM - A Robotic Aid to the Elderly for Mobility Assistance and Monitoring. In: *IEEE International Conference on Robotics and Automation* 2000, San Francisco, 570 –576.

[75] McDonald MJ: Active Research Topics in Human Machine Interfaces. Technical Report 2000. Sandia National Labs., Albuquerque, NM (US).

[76] Bueno L, Brunetti F, Frizzera A, Pons JL: Human-Robot Cognitive Interaction. In: *Wearable Robots: Biomechatronic Exoskeletons*. 1 ed. John Wiley & Sons, 2008 1:87-126.

[77] Hashimoto H, Sasaki A, Ohyama Y, Ishii C: Walker with hand haptic interface for spatial recognition. In *Proceeding of the ninth IEEE International Workshop on Advanced Motion Control* 2006, 311-316.

[78] Rentschler AJ, Simpson R, Cooper RA, Boninger ML: Clinical evaluation of Guido robotic walker. *Journal of Rehabilitation Research Development* 2008, 45(9):1281–1294.

[79] Shi F, Cao Q, Leng C, Tan H: Based On Force Sensing- Controlled Human-Machine Interaction System For Walking Assistant Robot. *Proceedings of the 8th World Congress on Intelligent Control and Automation* 2010, Jinan, China, 6528-6533.

[80] Zhou W, Xu L, Yang J: An Intent-Based Control Approach for an Intelligent Mobility Aid. *2nd International Asia Conference on Informatics in Control, Automation and Robotics* 2010, 54-58.

[81] McLachlan S, Arblaster J, Liu DK, Miro JV, Chenoweth L: A Multi-Stage Shared Control Method for an Intelligent Mobility Assistant. *Proceedings of the 2005 IEEE 9th International Conference on*

Rehabilitation Robotics, 426-429.

[82] Chuy O, Hirata Y, Kosuge K: Active Type Robotic Mobility Aid Control Based on Passive Behavior. Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems. San Diego, CA, USA, 165-170.

[83] Gharieb W: Intelligent Robotic Walker Design. International Conference on Automation, Robotics and Autonomous Systems, Sharm El Sheikh, Egypt, 2006.

[84] Lee G, Ohnuma T, Chong NY: Design and control of JAIST active robotic walker. Intel Serv Robotics, 125–135, 2010.

[85] Hirasawa M, Okada H, Shimojo M: The development of the plantar pressure sensor shoes for gait analysis. Journal Robot Mechatron 2007, 20(3):324–330.

[86] Wasson G, Gunderson J, Graves S, Felder R: An assistive robotic agent for pedestrian mobility. In Inter. Conf. on Autonomous Agents 2001, 169-173.

[87] Jiang Y, Wang S: Adapting Directional Intention Identification in Running Control of a Walker to Individual Difference with Fuzzy Learning. Proceedings of the 2010 IEEE International Conference on Mechatronics and Automation. Xi'an, China, 2010.

[88] Alwan M, Ledoux A, Wasson G, Sheth P, Huang C: Basic walker-assisted gait characteristics derived from forces and moments exerted on the walker's handles: Results on normal subjects. Medical Engineering & Physics 2007, 29:380–389, 2007.

[89] Patel M, Miro JV, Dissanayake G: Activity Recognition from the Interactions between an Assistive Robotic Walker and Human Users. HRI 2011.

[90] Frizera A, Gallego JA, Rocon E, Pons JL, Ceres R: Extraction of user's navigation commands from upper body force interaction in walker assisted gait. Biomedical Engineering Online 2010, 9(1):37.

[91] Abizanda P, Saro B, Sánchez P, Ceres R, Sánchez F, and Bonivardo G: Validación clínica del andador pseudorobótico de alta seguridad (asas). In Actas del III Congreso Iberoamericano sobre Tecnologías de Apoyo para la Discapacidad - IBERDISCAP 2004.

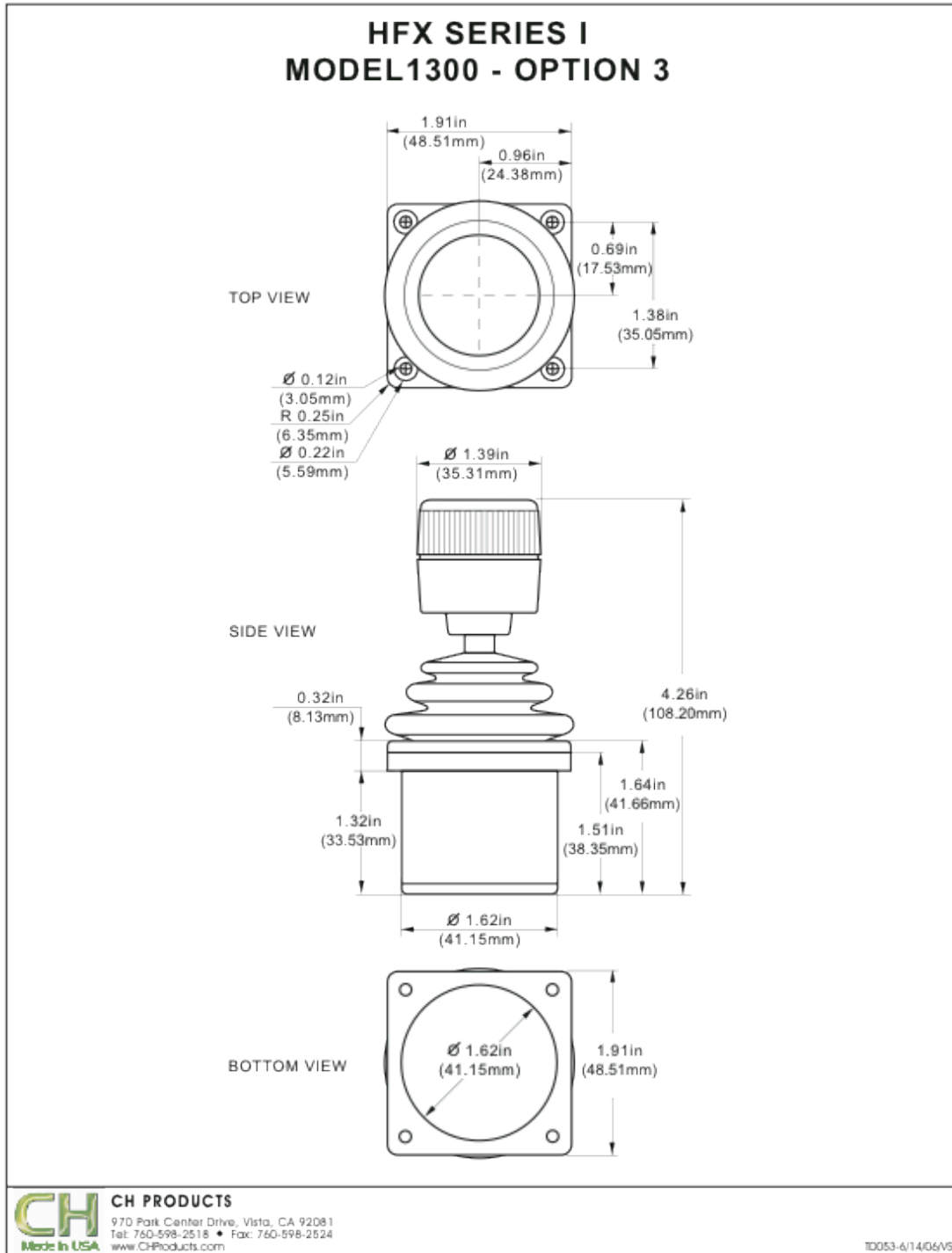
[92] Frizera A, Abellanas A, Ceres R, Pons JL, Raya R: Estudio y Caracterización de la Cinemática de los Pies en Marcha Asistida con Andadores. Revista Iberoamericana de Automática e Informática Industrial, v. 6, p. 74-84, 2009.

[93] Frizera A, Ceres R, Rocon E, Pons JL: Empowering and Assisting Natural Human Mobility: The Symbiosis Walker. *International Journal of Advanced Robotic Systems*, 2010.

- [94] Wright DL, Kemp TL: The dual-task methodology and assessing the attentional demands of ambulation with walking devices. *Phys Ther*, 72:306–312, 1992.
- [95] Hamzeh MA, Bowker P, and Sayegh A: The energy costs of ambulation using two types of walkers. *Clin Rehabil*, 2:119–23, 1998.
- [96] Arduino. www.arduino.cc. Accessed in 19 May 2010.
- [97] Bushberg JT., et al: *The Essential Physics of Medical Imaging*, (2e). Philadelphia: Lippincott Williams & Wilkins, pp.280, 2006.
- [98] Brown R, Hwang P: *Introduction to Random Signals and Applied Kalman Filtering*, 3rd. John Wiley & Sons, 1997.
- [99] Brookner E: *Tracking and Kalman Filtering Made Easy*. John Wiley & Sons, Inc, 1998.
- [100] Kosinski RA: Literature Review on Reaction Time. Clemson University.
- [101] Grewal M, Andrews A: (3rd) *Kalman Filtering – Theory and Practice using Matlab*. John Wiley & Sons.
- [102] Welch G, Bishop G: *An Introduction to the Kalman Filter*. UNC-Chapel Hill, 2006.
- [103] Benedict TR, Bordner GW: Synthesis of an Optimal track-while-scan smoothing equations. *IRE Transactions Automation Control*, 7, pp. 27-32, 1962.
- [104] Maybeck P: *Stochastic Models, Estimation and Control*. vol. 141. *Academic Press, Inc*, 1979.
- [105] Rocón E, Ruiz J, Moreno J, Pons J, Miranda A: Tremor Characterization: Algorithms for the study of tremor time series. *Sensors*, 2010.
- [106] Lee SH, Howlett RJ, Walters SD: Fuzzy and Neuro-fuzzy Techniques for Modelling and Control. *KES 2006*, pp.1206-1215, 2006.
- [107] Zadeh LA: Fuzzy Sets, *Information Control* 8, pp. 338-353, 1965.
- [108] Shing J, Jang R, Sun C: Neuro-Fuzzy Modeling and Control. *Proceedings of the IEEE*, vol. 83, no 3, pp.381, 1965.
- [109] Lee K: Fuzzy Inference neural network for fuzzy model tuning. *IEEE Trans. Syst. Man, Cybern.*, 26(4B), pp.637-645, 1996.
- [110] George K, Ute SC, Bo Y: *Fuzzy Set Theory*. Prentice Hall, 1997.

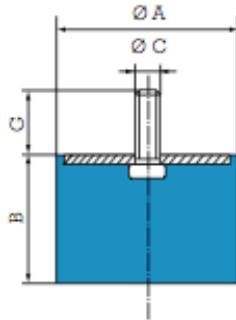
Appendix A

A.1. Joystick and springs Datasheet

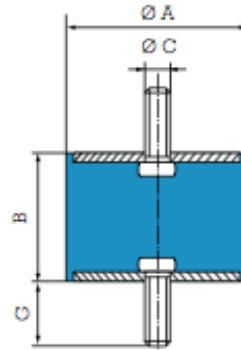


DIMENSIONS AND COMPRESSIVE LOADS

SINGLE STUD FIXING



DOUBLE STUDS FIXING



New RADIAFLEX references

Ø A mm	B mm	Ø C	G mm	Compression		Ref.
				Max. load daN	Deflection mm	
12.5	10	M3	10	12	2	511110
	13.5			11	2.5	511128
	15			10	3	511115
	20			8	3.5	511125
16	10	M4	10	20	2	511150
	15			3	511151	
	10	M5	12	20	2	511292
	15			3	511294	
	20			4	511296	
25	5	511298				
20	8.5	M6	16.5	40	1.5	511200
	15			4	511215	
	20			5	511220	
	25			5.5	511225	
	30			7	511230	
25.5	10	M6	18	80	2	511158
	15			3.5	511158	
	20			5	511159	
	30			6	511160	
	10			M8	20	80
15	3.5	511270				
19	4.5	511251				
22	5.5	511275				
25	6	511280				
30	6	511285				
40	10	511290				
30	15	M8	25	90	3.5	511308
	22			6	511310	
	30			8	511312	
	40			9	511314	
	30			M8	20	120
40	10	511161				
40	20	M10	25	160	5	511450
	25			6	511401	
	35			6	511401	
	35			6	511452	

See Vibrachoc elastomer range : Threaded studs

Ø A mm	B mm	Ø C	G mm	Compression		Shear*		Ref.
				Max. load daN	Deflect mm	Max. load daN	Deflect mm	
10	8	M3	6	10	1.6	1.25	0.9	**
12	8	M3	6	12	1.2	1.5	0.75	**
12.5	10	M5	10	12	2	1.5	1.5	521293
	15			3	2.5	2	521128	
	20			3.5	2.5	4	521295	
16	10	M4	10	20	1.5	2.5	1.5	521650
	15			3	2	521651		
	10	M5	12	20	1.5	2.5	1.5	521292
	15			3	2	521294		
20	4			4	521296			
25	5	5	521298					
20	8.5	M6	16.5	40	0.6	5	5	521178
	15			4	2.5	521249		
	20			4.5	3.5	521297		
	25			5	4.5	521299		
	30			5	4.5	521319		
25.5	10	M6	18	80	1.5	8	1.5	521655
	15			2.5	2.5	521656		
	20			2.5	4	521652		
	30			7.5	6	521653		
	10			M8	20	80	1.5	8
15	2.5	2.5	521341					
19	4	4	521251					
22	5.5	4.5	521342					
25	6	6	521343					
30	10	6.5	6	521344				
30	15	M8	25	90	3	11	2.5	521306
	22			6	4	521310		
	30			8	11	6	521312	
	40			9	11	7.5	521314	
	30			M8	20	150	6	20
40	10	20	7.5			521657		
40	20	M10	25	160	4	20	3	521450
	25			6	20	5.5	521401	
	35			6	20	6.5	521452	
	40			10	20	7.5	521454	
	40			10	20	7.5	521454	
	40			11	20	9	521455	

A.2. Y-signal Circuit Hardware

3 main blocks constitute the circuit: Amplification plus Offset block, Low pass filter block and Amplification block.

The Amplification + Offset block (1) is compound by an inverting sum amplifier (TL084) using one operation amplifier.

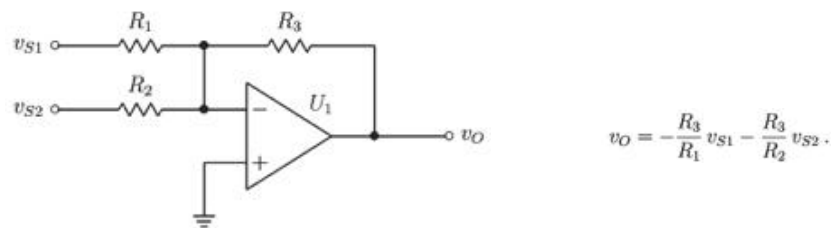


Figure A. 1 Inverting sum amplifier the respective gain equation.

The input v_{s1} corresponds to the raw Y signal acquired with the joystick. The input v_{s2} corresponds to -2.5 V, which is the offset that Y brings from the joystick that will be summed to the signal eliminating its offset.

It was chose to amplify the signal with a gain of 2. So the output (v_o) was calculated with the expression of the figure A.1 and for this R_1 and R_2 are both $2\text{k}\Omega$ and R_3 are $3.9\text{ k}\Omega$.

The result will be a signal 2x amplified and inverted.

The low pass filter block (2) is to attenuate the frequencies that are above a cut-off frequency of 1Hz.

The filter used was a filter of 2nd order Salley-Key with Butterworth approximation (TL084). In figure A.2 is represented the low-pass filter and the respective equations.

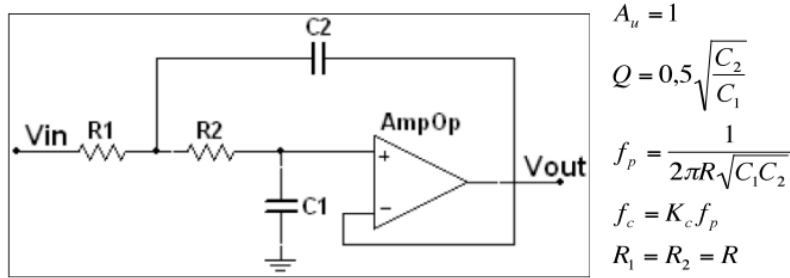


Figure A. 2 Low pass Filter of 2nd order Sallen-Key and the respective equations. A_u is the bandwidth gain, Q is the quality, and f_c is the cut-off frequency.

As it desired to have a Butterworth response the Q value should be 0.707 and K_c equal to 1. As in this stage is not wanted to amplify the signal, A_u should be 1.

Using the expressions presented in figure A.2 the components used in the design of the low-pass filter were: R_1 and R_2 equal to 10k Ω , C_1 equal to 0.6pF and C_2 equal to 1.2pF.

The Amplification block (3) is a non-inverting amplifier (LM741) presented in figure A.3. This block aims to amplify the signal with a gain of 1.5. Thus, the total gain of the circuit is 3.5, amplifying the signal to a range of [0,3.3].

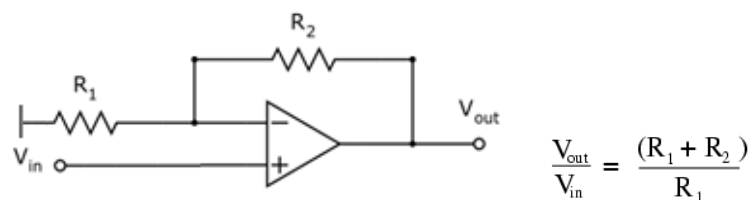


Figure A. 3 Non-inverting Amplifier and the respective gain equation.

Following the equation in figure 3 to obtain a gain of 1.5, R_1 is equal to 6.8 k Ω and R_2 is equal to 3.6 k Ω .

Between elements (1)-(2), and (2)-(3) it was integrated a Voltage Buffer amplifier (LM741), shown in figure A.4.

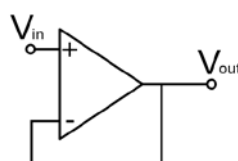


Figure A. 4 Voltage Buffer amplifier.

A Voltage Buffer amplifier is used to transfer a voltage from a first circuit, having a high output impedance level, to a second circuit with a low input impedance level. The interposed buffer amplifier prevents the second circuit from loading the first circuit unacceptably and interfering with its desired operation. In the ideal voltage buffer, the input resistance is infinite, the output resistance zero (impedance of an ideal voltage source is zero). The voltage will be transferred unchanged so this is a unity gain buffer.

The completed circuit is presented in figure A.5.

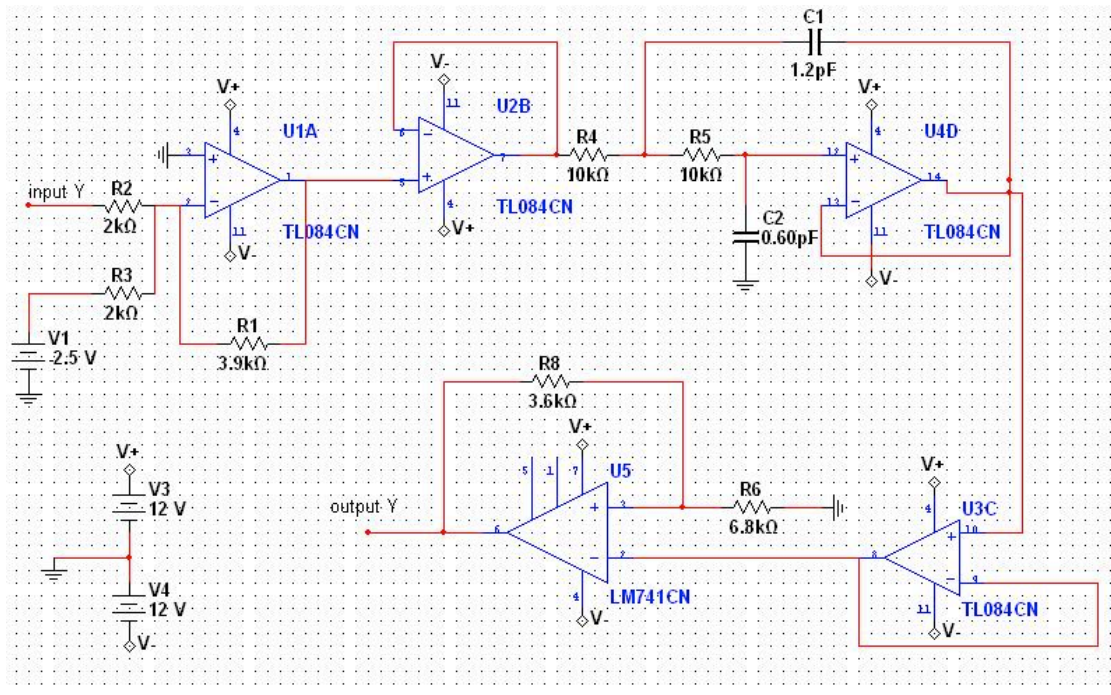


Figure A. 5 Complete Y-signal acquisition and processing circuit.

A.3. Z-signal circuit

3 main blocks constitute the circuit: Amplification plus Offset block, Low pass filter block and Amplification block.

The Amplification + Offset block (1) is also an inverting sum amplifier using one operation amplifier (TL084). The aim is the same as previously, which is to take away the offset of the Z signal, which value is around 2.5V and then amplify the sum.

The input v_{s1} corresponds to the raw Z signal acquired with the joystick and the input v_{s2} corresponds to -2.5 V, which corresponds to the offset of the Z signal. A gain of 5 was used to

amplify the Z signal. Using the expression of the figure A.1, R_1 and R_2 are both $3k\Omega$ and R_3 are $15k\Omega$.

The low pass filter block (2) was the same as used with the signal Y.

The Amplification + Offset block (3) is compound by two amplifiers. First, a non-inverting amplifier (TL084) with a gain of 2, where R_1 and R_2 are both $3k\Omega$. Then, an inverting sum amplifier (LM741) with a gain of 1, where R_1 , R_2 and R_3 are all $3k\Omega$. And in this case it is summed an offset of $1.6V$. With the Z signal it is possible to distinguish the user's intentions to turn right or left. This distinction is made between a negative or positive sign of the signal. Since the Arduino platform doesn't read negative signals, it is needed to sum an offset to the signal. This offset of $1.6V$ is half the maximum voltage that the Arduino can read, which is $3.3V$.

The total gain of the circuit is 7, putting the signal into a range of $[0,3.3]$.

Between elements (1)-(2), and (2)-(3) it was integrated a Voltage Buffer amplifier, shown in figure A.4.

The completed circuit is presented in figure A.6.

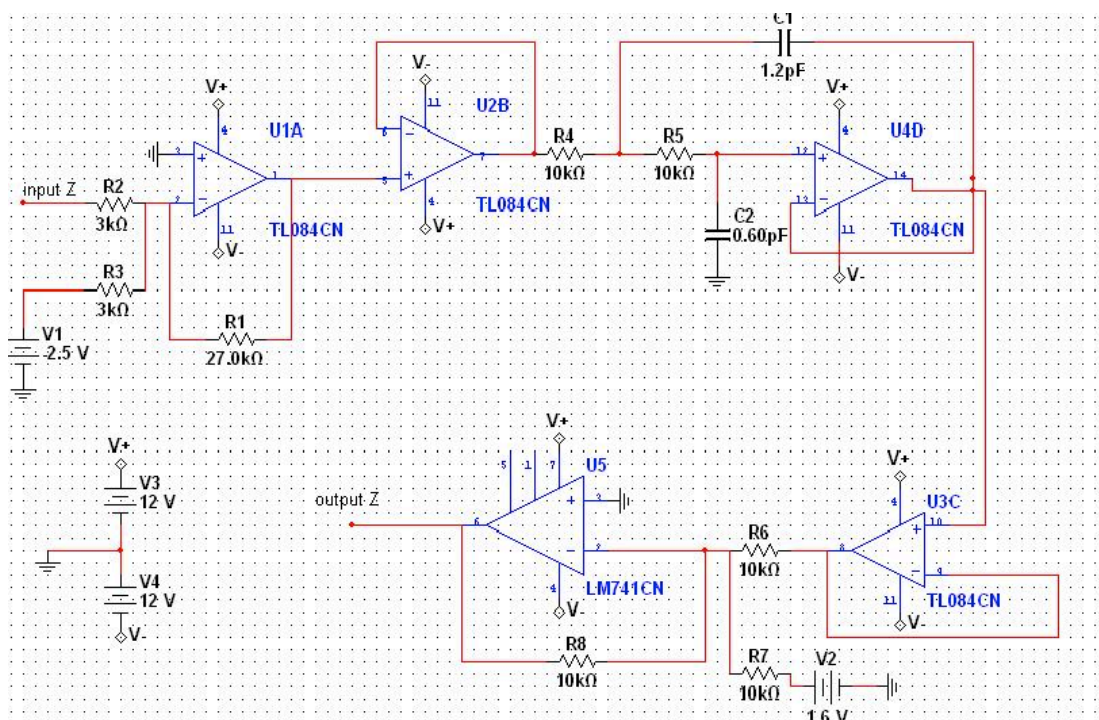


Figure A. 6 Complete Z-signal acquisition and processing circuit.

Appendix B

B.1. Sharp Sensor Datasheet

SHARP

GP2Y0A21YK/GP2Y0D21YK

GP2Y0A21YK/ GP2Y0D21YK

General Purpose Type Distance Measuring Sensors

■ Features

1. Less influence on the color of reflective objects, reflectivity
2. Line-up of distance output/distance judgement type
 Distance output type (analog voltage) : **GP2Y0A21YK**
 Detecting distance : 10 to 80cm
 Distance judgement type : **GP2Y0D21YK**
 Judgement distance : 24cm
 (Adjustable within the range of 10 to 80cm [Optionally available])
3. External control circuit is unnecessary
4. Low cost

■ Applications

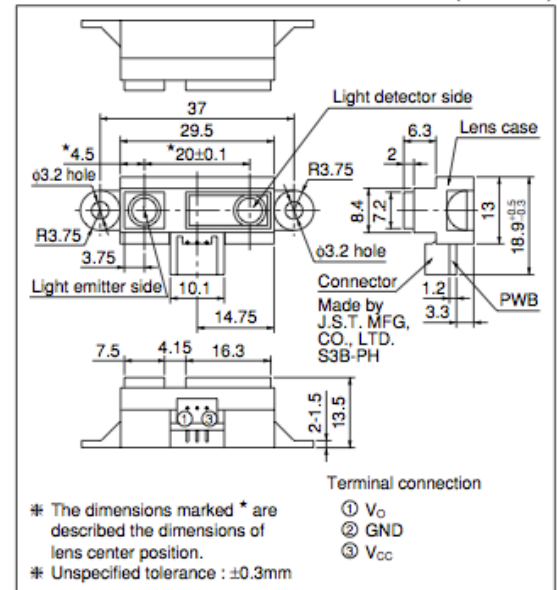
1. TVs
2. Personal computers
3. Cars
4. Copiers

■ Absolute Maximum Ratings $(T_c=25^{\circ}\text{C}, V_{CC}=5\text{V})$

Parameter	Symbol	Rating	Unit
Supply voltage	V_{CC}	-0.3 to +7	V
Output terminal voltage	V_O	-0.3 to $V_{CC}+0.3$	V
Operating temperature	T_{opr}	-10 to +60	$^{\circ}\text{C}$
Storage temperature	T_{stg}	-40 to +70	$^{\circ}\text{C}$

■ Outline Dimensions

(Unit : mm)



■ Recommended Operating Conditions

Parameter	Symbol	Rating	Unit
Operating supply voltage	V_{CC}	4.5 to +5.5	V

■ Electro-optical Characteristics

($T_c=25^\circ\text{C}$, $V_{CC}=5\text{V}$)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Distance measuring range	ΔL	*1 *3	10	-	80	cm	
Output terminal voltage	GP2Y0A21YK	V_O	$L=80\text{cm}^{*1}$	0.25	0.4	0.55	V
	GP2Y0D21YK	V_{OH}	Output voltage at High *1	$V_{CC}-0.3$	-	-	V
	GP2Y0D21YK	V_{OL}	Output voltage at Low *1	-	-	0.6	V
Difference of output voltage	GP2Y0A21YK	ΔV_O	Output change at $L=80\text{cm}$ to 10cm^{*1}	1.65	1.9	2.15	V
Distance characteristics of output	GP2Y0D21YK	V_O	*1 *4 *2	21	24	27	cm
Average Dissipation current	I_{CC}	$L=80\text{cm}^{*1}$	-	30	40	mA	

Note) L : Distance to reflective object

*1 Using reflective object : White paper (Made by Kodak Co. Ltd. gray cards R-27 - white face, reflective ratio : 90%)

*2 We ship the device after the following adjustment : Output switching distance $L=24\text{cm}\pm 3\text{cm}$ must be measured by the sensor

*3 Distance measuring range of the optical sensor system

*4 Output switching has a hysteresis width. The distance specified by V_O should be the one with which the output L switches to the output H

Fig.1 Internal Block Diagram

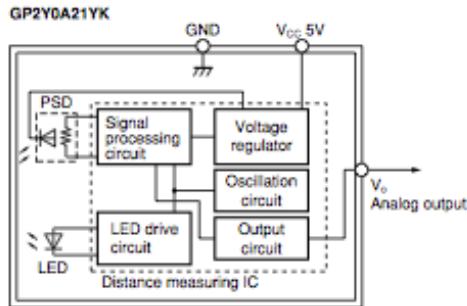


Fig.2 Internal Block Diagram

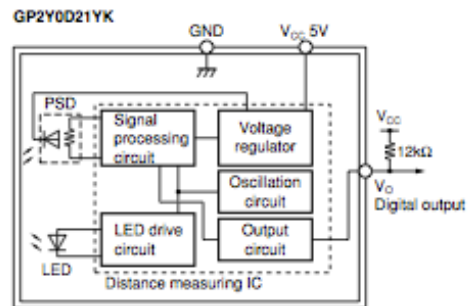


Fig.3 Timing Chart

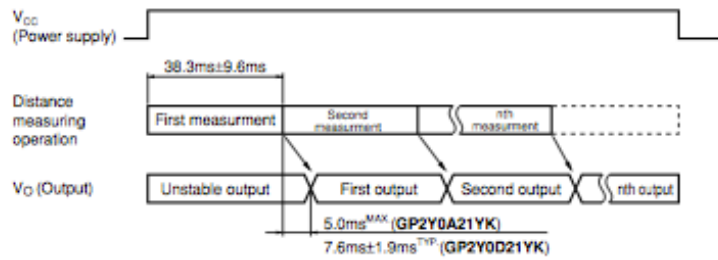


Fig.4 Distance Characteristics

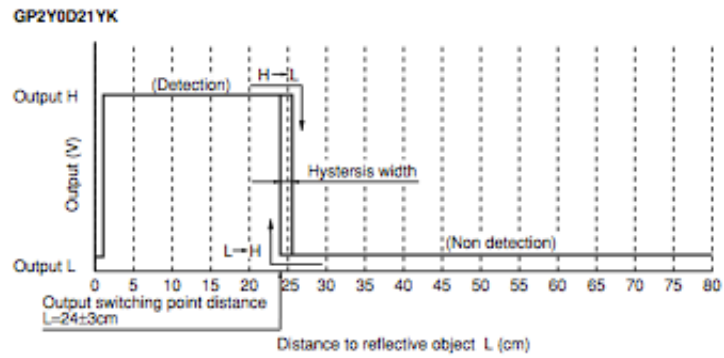
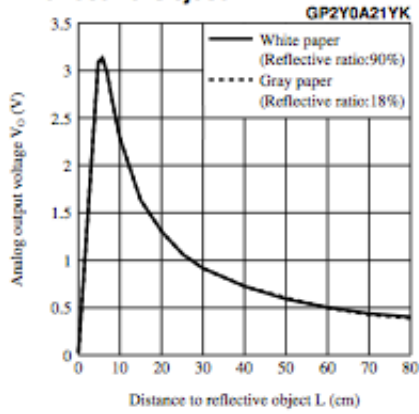


Fig.5 Analog Output Voltage vs. Distance to Reflective Object



B.2. Sensor Operation

The sensor uses triangulation and a small linear CCD array to compute the distance and/or presence of objects in the field of view. The basic idea is to emit a pulse of IR light by the emitter. This light travels out in the field of view and either hits an object or just keeps on going. In the case of no object, the light is never reflected and the reading shows no object. If the light reflects off an object, it returns to the detector and creates a triangle between the point of reflection, the emitter, and the detector (Figure B.1).

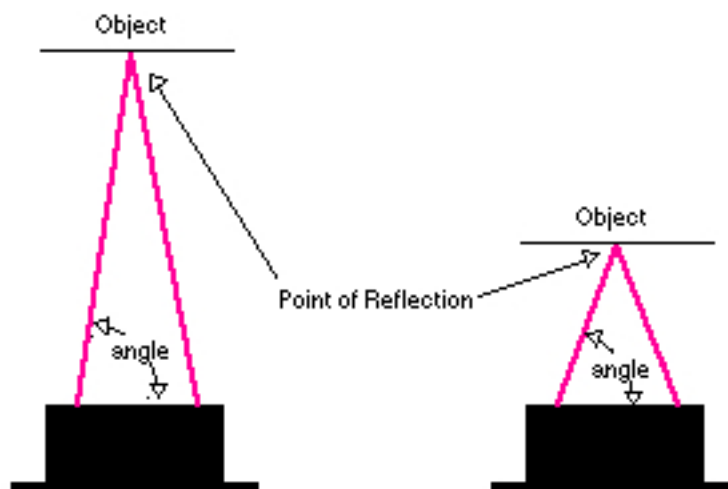


Figure B. 1 Scheme of the Sensor operation.

The angles in this triangle vary based on the distance to the object. The receiver portion of these new detectors is actually a precision lens that transmits the reflected light onto various portions of the enclosed linear CCD array based on the angle of the triangle described above. The CCD array can then determine what angle the reflected light came back at and therefore, it can calculate the distance to the object.

This new method of ranging is almost immune to interference from ambient light and offers amazing indifference to the colour of object being detected.

The graph showed in figure B.2 represents the sensor operation curve to different colours, textures and light.

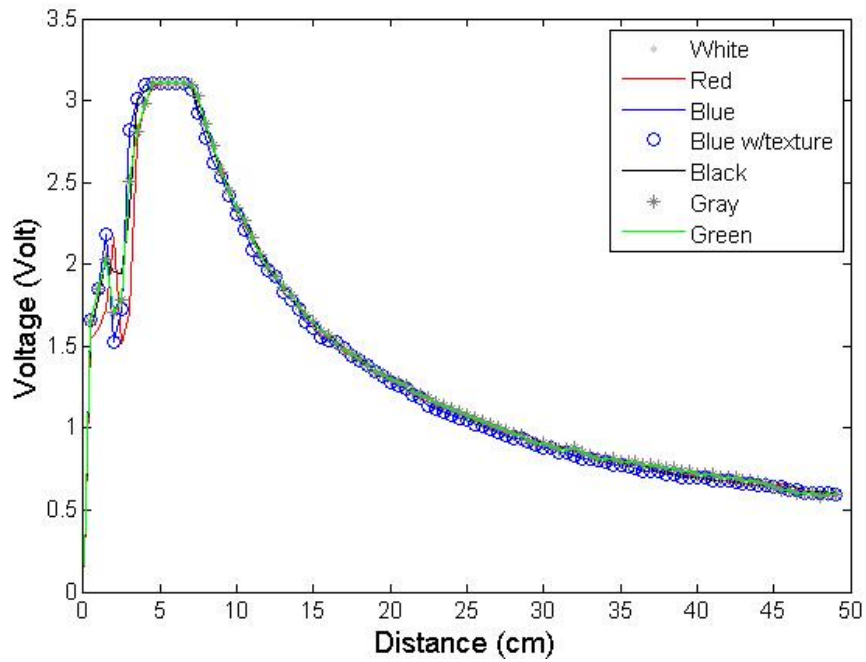


Figure B. 2 IR sensor operation curve.

The experiments showed that after 7 cm the sensor has a output response (distance vs Voltage) that can be quantified as a power function: $y=16,64.x^{0,8557}$, $R= 0,9994$. It is also verified that the response is not altered by the colour, texture and light. Thus, the sensor response has the following characteristics:

Table B. 1 Maximum and Minum values of the Voltage/distance that the sensor can present/detect.

Maximum Voltage	3,2V
Minimum Voltage	0 V
Maximum Distance	80 cm
Minimum Distance	6.5 cm

Appendix C

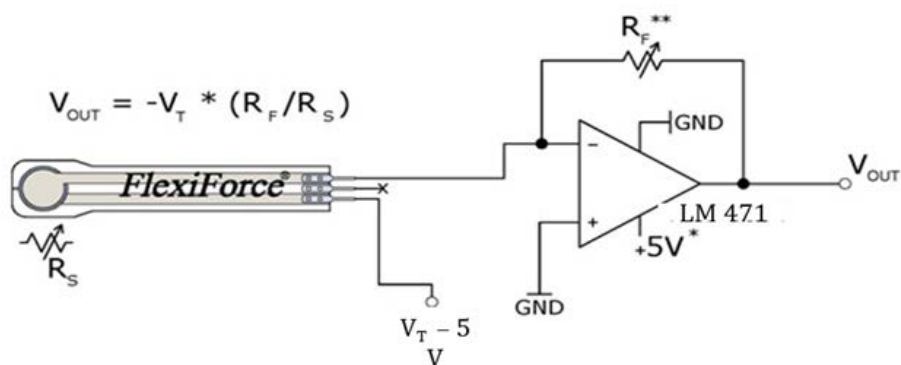
C.1. Flexiforce Properties

SENSOR PROPERTIES

STANDARD FLEXIFORCE SENSOR (MODEL A201)

Sensor Properties	
Thickness	0.008 (0.208 mm)
Length	8" (203 mm) 6" (152 mm) 4" (102 mm) 2" (51 mm)
Width	0.55" (14 mm)
Sensing Area	0.375" (9.53 mm) diameter
Connector	3-pin male square pin (center pin is inactive)
Typical Performance	
Force Ranges	0-1 lb (4.4 N) 0-25 lbs (110 N) 0-100 lbs (440 N)*
Operating Temperature Range	15°F to 140°F (-9°C to 60°C)
Linearity (Error)	+/- 3%
Repeatability	+/- 2.5% of full scale (conditioned sensor, 80% force applied)
Hysteresis	<4.5% of full scale (conditioned sensor, 80% force applied)
Drift	<5% per logarithmic time scale (constant load of 90% sensor rating)
Response Time	<5 microseconds
Output Change/Degree F	Up to 0.2% (~0.36% / °C). Loads <10 lbs, operating temperature can be increased to 165°F (74°C).

C.2. Acquisition circuit of the flexiforce signal.

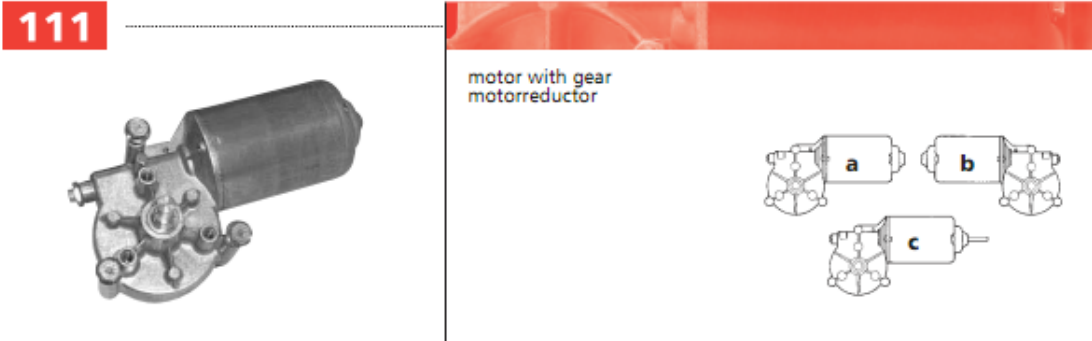


- * Supply Voltages should be constant
- ** Reference Resistance R_F is 1kΩ to 100kΩ
- Sensor Resistance R_S at no load is >5MΩ
- Max recommended current is 2.5mA

Figure C. 1 Schematic of the acquisition circuit for the FlexiForce signal.

Appendix D

D.1 Motor's specifications (DOGA: 111.9041.30.00)



Reference number Referencia	Nominal voltage Voltaje nominal	Nominal torque Par nominal	Nominal speed Velocidad nominal	Nominal current Corriente nominal	Starting torque Par de arranque	Starting current Corriente de arranque	Shaft Eje	Connectors Conectores	Wiring (kg) Esp. delc.	Transmission ratio Relación de reducción	Approximate weight Peso aproximado	Waterproofness Grado de estanqueidad	Wheel material Material rueda	Design a/b/c Diseño a/b/c	Case Caja
	U _n (V)	M _n (N.m. / lbf.in)	n _n (r.p.m.)	I _n (A)	M _a (N.m. / lbf.in)	I _a (A)				i	P (kg / lb.t)	IP			
111.3711.20.00	12	5 / 44.2	40	5	25 / 221.2	25	E22	C25	EE2	62:1	1.25 / 3.34	IP53	PLA	a	1
111.3711.30.00	24	5 / 44.2	40	2.5	25 / 221.2	13	E22	C25	EE2	62:1	1.25 / 3.34	IP53	PLA	a	1
111.3761.20.00	12	5 / 44.2	40	5	25 / 221.2	25	E23	C25	EE2	62:1	1.25 / 3.34	IP53	PLA	a	1
111.3761.30.00	24	5 / 44.2	40	2.5	25 / 221.2	13	E23	C25	EE2	62:1	1.25 / 3.34	IP53	PLA	a	1
111.3761.20.00E	12	5 / 44.2	40	5	25 / 221.2	25	E23	C25	F2	62:1	1.25 / 3.34	IP53	PLA	a	1
111.3761.30.00E	24	5 / 44.2	40	2.5	25 / 221.2	13	E23	C25	F2	62:1	1.25 / 3.34	IP53	PLA	a	1
111.3763.20.00	12	6 / 53.1	25	4	25 / 221.2	15	E23	C25	EE2	62:1	1.25 / 3.34	IP53	PLA	a	3
111.3763.30.00	24	6 / 53.1	25	2	25 / 221.2	8	E23	C25	EE2	62:1	1.25 / 3.34	IP53	PLA	a	3
111.4761.30.00	24	5 / 44.2	40	2.5	25 / 221.2	13	E23	C25	EE2	62:1	1.25 / 3.34	IP53	PLA	b	1
111.9031.20.00	12	3 / 26.5	70	6	25 / 221.2	34	E23	C25	EE2	62:1	1.25 / 3.34	IP53	PLA	a	2
111.9031.30.00	24	3 / 26.5	70	3	25 / 221.2	17	E23	C25	EE2	62:1	1.25 / 3.34	IP53	PLA	a	2
111.9039.30.00	24	1.5 / 13.2	240	4	14 / 123.9	23	E23	C26	EE1	49:4	1.25 / 3.34	IP53	PLA	a	4
111.9041.30.00	24	5 / 44.2	40	2.5	25 / 221.2	13	E24	C25	EE2	62:1	1.30 / 3.48	IP53	BRO	a	1
111.9094.20.00	12	5 / 44.2	40	5	25 / 221.2	25	E52	C2	EE2	62:1	1.25 / 3.34	IP53	PLA	a	1
111.9107.30.00	24	1.5 / 13.2	240	4	14 / 123.9	23	E24/E53	C26	EE1	49:4	1.25 / 3.34	IP40	CEL	c	4
111.9114.30.00	24	3 / 26.5	70	3	25 / 221.2	17	E24/E54	C37	EE2	62:1	1.25 / 3.34	IP40	PLA	c	2
111.9199.30.00	24	3 / 26.5	100	3	20 / 177.01	24	E24	C26	F3	59:2	1.25 / 3.34	IP53	PLA	a	59

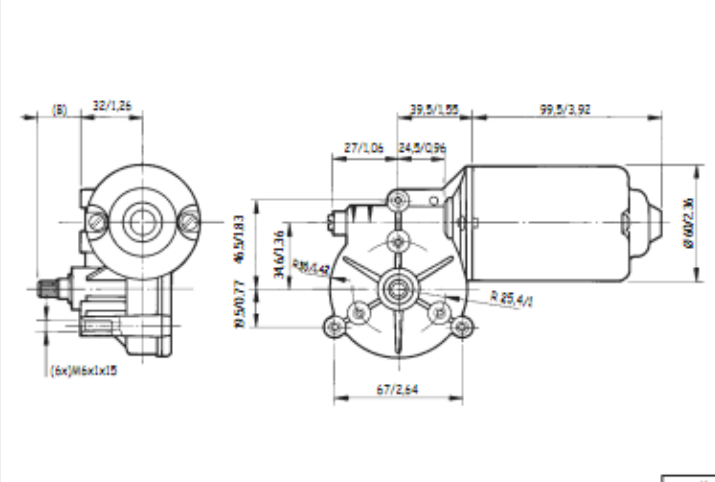
DOGA can develop for you on this series a customised version:
DOGA puede desarrollar para usted una versión a medida:

▼

SELF-LOCKING / AUTO-BLOQUEO
BRONZE WHEEL / RUEDA DE BRONCE
RESINBONDED WHEEL / RUEDA DE CELOTEX
FRONT & REAR SHAFT / EJE DELANTERO Y POSTERIOR
HALL SENSOR / SENSOR HALL

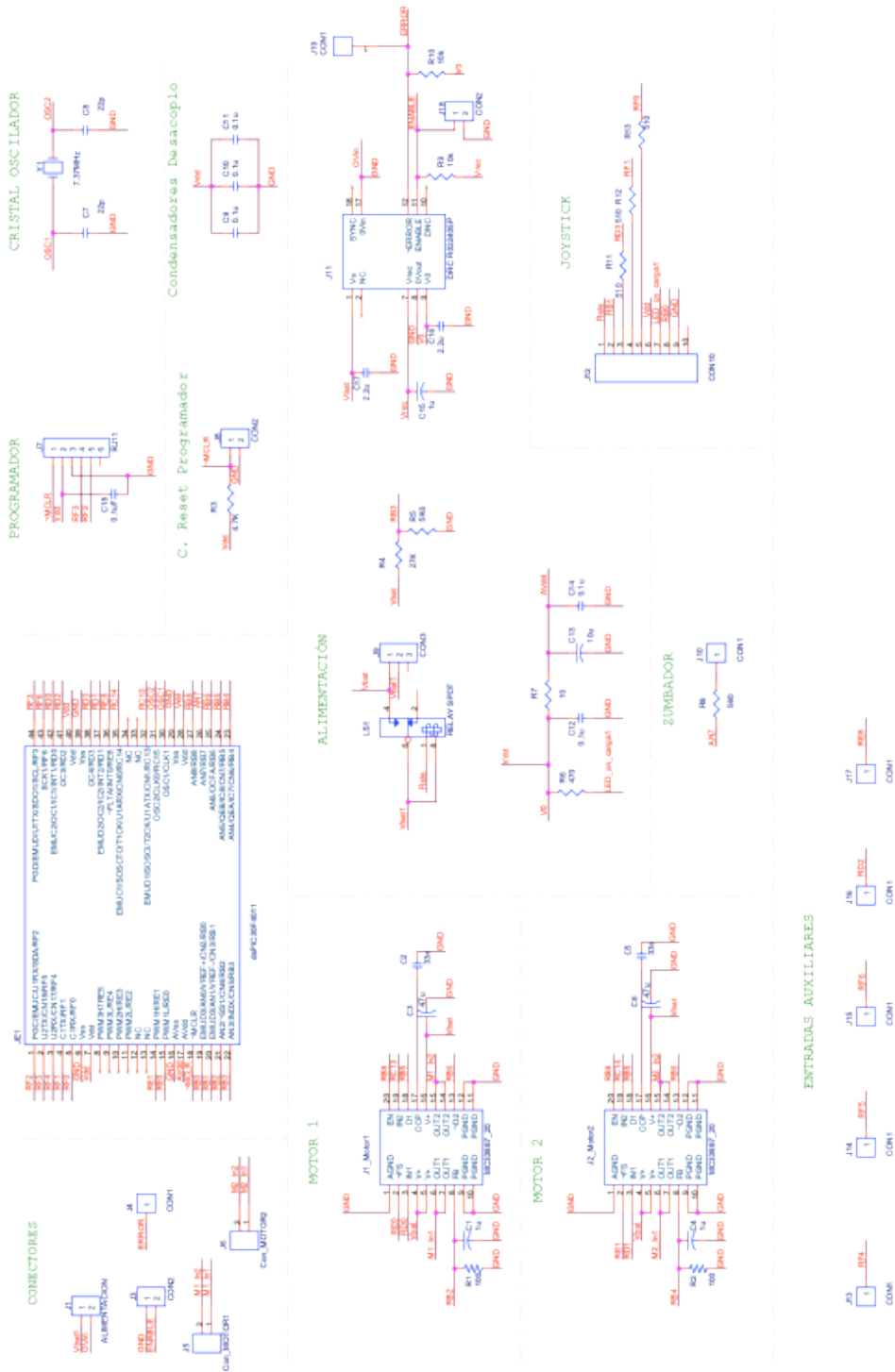
▲

and many more / y mucho más
please contact us / por favor contactenos



mm / in

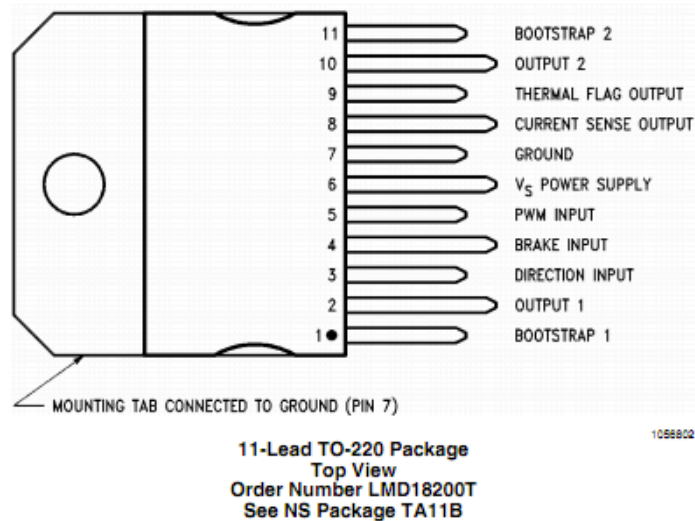
D.2. Control board used in the NeoASAS platform to control the motors



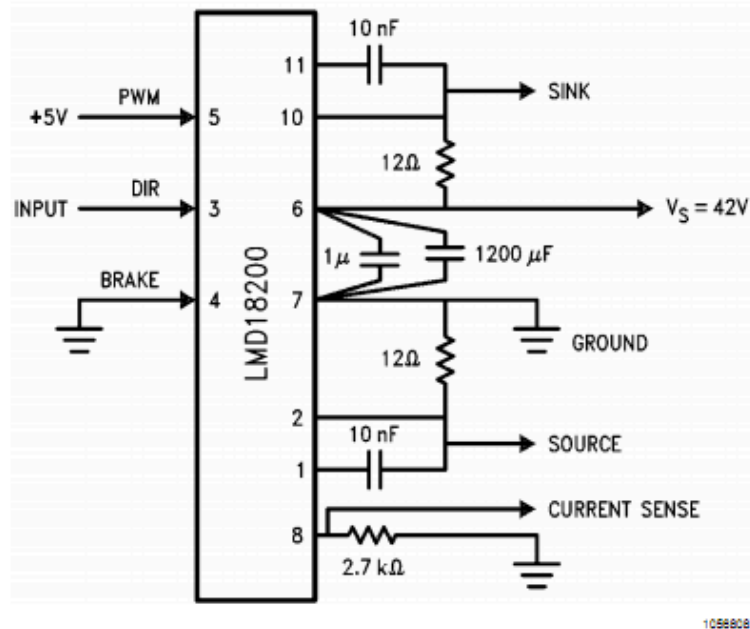
D.3. Control platform used in the ASBGo walker to control the motors

D.3.1 H-bridges (LMD18200, 3A, 55V)

Connection Diagram and Ordering Information



Test Circuit



Absolute Maximum Ratings *(Note 1)*

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Total Supply Voltage (V_S , Pin 6)	60V
Voltage at Pins 3, 4, 5, 8 and 9	12V
Voltage at Bootstrap Pins (Pins 1 and 11)	$V_{OUT} + 16V$
Peak Output Current (200 ms)	6A
Continuous Output Current <i>(Note 2)</i>	3A

Power Dissipation <i>(Note 3)</i>	25W
Power Dissipation ($T_A = 25^\circ C$, Free Air)	3W
Junction Temperature, $T_{J(max)}$	150°C
ESD Susceptibility <i>(Note 4)</i>	1500V
Storage Temperature, T_{STG}	-40°C to +150°C
Lead Temperature (Soldering, 10 sec.)	300°C

Operating Ratings *(Note 1)*

Junction Temperature, T_J	-40°C to +125°C
V_S Supply Voltage	+12V to +55V

Table D. 1 Logic True table

PWM	Dir	Brake	V out (sink)	Vout (source)	Right Motor	Left Motor
H	H	L	0	12	Backwards	Forwards
H	L	L	12	12 (50%*)	Forwards	Backwards
L	X	L	12	12	Stop	Stop
H	H	H	12 (75%*)	12 (75%*)	Stop	Stop
H	L	H	12 (25%*)	12 (25%*)	Stop	Stop
L	X	H	12 (25%*)	12 (25%*)	Stop	Stop

*Square wave duty cycle
H-High; L-Low; X-off

Scientific Output Indicators

From this thesis resulted:

- The partnership with Orthos XXI, a company of orthopedic products;
- The application for a PhD;
- The application for a FCT Project;
- International Conferences articles:
 - Martins, M., Frizera, A., Santos, C., Ceres, R. 2010. Classificação de Dispositivos de Assistência à marcha, IBERDISCAP, Palma de Maiorca. (accepted - Oral Presentation);
 - Martins, M., Frizera, A., Santos, C., Ceres, R. 2010. Review and Classification on Human Gait Training and Rehabilitation Devices. AAATE'11, Maasctrich. (accepted - Oral Presentation);
 - Martins, M., Frizera, A., Santos, C., Urendes, E., Ceres, R. 2011. A Novel Human-Machine interface for guiding Smart Walkers. The NeoASAS Purpose.IEEE-2012 ISSNIP Biosignals and Biorobotics for better and safer living Conference. (accepted- Oral Presentation);
 - Martins, M., Frizera., A., Santos, C., Ceres, R. 2012. Online control of Smart Walker through a monitoring system to infer the user's command intentions. 7th ACM/IEEE International Conference on Human Robot Interaction. (submitted);
 - Martins, M., Frizera., A., Santos, C., Ceres, R. 2012. Smart Walker control through the inference of the user's command intentions. International Conference in Robotics and Automation, Minnesota, USA. (submitted);
- Journal articles:
 - Martins, M., Santos, C., Frizera, A., Ceres, R.. Assistive Mobility Devices focusing on Smart Walkers: Classification and Review. Robotics and Autonomous System.(Under Review)
 - Martins, M., Santos, C., Frizera, A., Ceres, R. Online Control of an assistive device through user's command intentions derived by a novel interface integrated on the ASBGo walker. IEEE Transactions on Neural Systems and Rehabilitation Engineering. (submitted);