

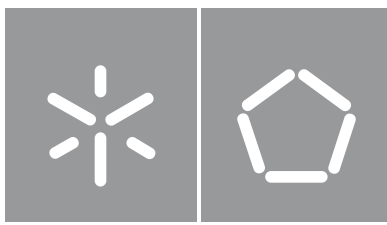


Sara Margarida Alves Monteiro

**Human-in-the-loop control for lower limb
assistance driven by exoskeleton**

Universidade do Minho
Escola de Engenharia





Universidade do Minho

Escola de Engenharia

Sara Margarida Alves Monteiro

**Human-in-the-loop control for lower limb
assistance driven by exoskeleton**

Dissertação de Mestrado

Mestrado Integrado em Engenharia Biomédica

Ramo Eletrónica Médica

Trabalho efetuado sob a orientação de

Doutora Joana Figueiredo

Professora Doutora Cristina Santos

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

Licença concedida aos utilizadores deste trabalho

[Caso o autor pretenda usar uma das licenças Creative Commons, deve escolher e deixar apenas um dos seguintes ícones e respetivo lettering e URL, eliminando o texto em itálico que se lhe segue. Contudo, é possível optar por outro tipo de licença, devendo, nesse caso, ser incluída a informação necessária adaptando devidamente esta minuta]



Atribuição-NãoComercial-SemDerivações
CC BY-NC-ND

<https://creativecommons.org/licenses/by-nc-nd/4.0/>

Agradecimentos

Gostaria de começar por agradecer aos meus pais, que sempre me encorajaram e apoiaram ao longo desta jornada, que foram os melhores mentores que eu poderia ter pedido, e que me aturaram durante todos estes anos, sempre com uma grande paciência. Ao meu irmão, um dos meus melhores amigos e companheiros que a vida me deu, um grande obrigada. Agradeço também a toda a minha família pelo apoio incondicional e preocupação demonstrada, e também pela confiança constante que depositaram em mim.

Quero também agradecer a todos os amigos que me apoiaram imenso, e estiveram sempre comigo nos bons e maus momentos, ao longo dos últimos cinco anos. Vocês tornaram-se numa segunda família, e não poderia ter pedido melhor companhia, tanto para os momentos de estudo como os de divertimento. Um agradecimento especial à Bruna, à Catarina, ao Gil, ao João, ao Jorge, ao José, ao Rúben, e à Sara, pelo companheirismo incondicional que demonstraram, e aos meus dois companheiros de residência, Daniel e Manuel, pela bonita amizade que formamos.

Por fim, e não menos importante, gostaria de expressar uma profunda gratidão à minha orientadora, Doutora Joana Figueiredo, que ao longo deste último ano me apoiou e ajudou imenso, e cuja orientação foi o alicerce deste trabalho. Agradeço também à Professora Doutora Cristina Santos, pela oportunidade que me deu em trabalhar neste projeto extraordinário e pela confiança que depositou em mim. Agradeço a todos os meus colegas e companheiros do BiRD Lab, por toda a vossa ajuda e disponibilidade demonstrada. Por fim, agradeço ao laboratório LABIOMEF, da Faculdade de Desporto da Universidade do Porto, pela colaboração na realização deste projeto.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

Resumo

Atualmente, as lesões musculoesqueléticas relacionadas com o trabalho (LMERTs) afetam aproximadamente 60% dos trabalhadores europeus (1). Permanecer de pé durante longos períodos e carregar e levantar objetos pesados são atividades que introduzem uma carga significativa nas pernas dos trabalhadores (1). Nem todos os processos industriais podem ser completamente automatizados e, portanto, é ainda necessário que os trabalhadores executem tarefas repetitivas (2). Os exoesqueletos podem ser utilizados para aumentar a capacidade física e diminuir a tensão física nos músculos dos utilizadores, contudo, os exoesqueletos atuais ainda não sincronizam os seus movimentos com a intenção do humano nem assistem cada trabalhador de forma individualizada (2).

O principal objetivo desta dissertação foi o desenvolvimento e validação de um controlo *human-in-the-loop* (HITL) para um exoesqueleto de membros inferiores. O propósito do controlo foi a otimização, em tempo-real, do perfil de torque do exoesqueleto, para um perfil que minimizasse o esforço do utilizador, avaliado pelo seu custo metabólico (CM) e torque de interação.

Primeiramente, um modelo regressor capaz de estimar o CM de uma pessoa, em tempo-real, com base em dados de quatro sensores inerciais foi desenvolvido e integrado no exoesqueleto. O modelo foi treinado com dados de três atividades: (i) estar de pé; (ii) andar; e (iii) estar sentado. Durante uma validação experimental, o modelo alcançou um erro quadrático médio de 0.66 W/kg.

De seguida, foi desenvolvido um controlo de torque que permitiu a manipulação dos atuadores do dispositivo para seguirem um perfil de torque gerado por interpolação polinomial. Este foi formado por um controlador de nível médio que estima a fase da marcha e obtém a referência de torque, e um controlador de nível baixo usado para movimentar o atuador de acordo com a referência e fase da marcha. Uma validação conceptual do controlo mostrou que este foi capaz de assistir o utilizador com sucesso.

Por fim, um controlo HITL foi desenvolvido, pela integração do modelo regressor e o controlo de torque com um algoritmo otimizador. O otimizador adaptou o perfil de torque, em tempo-real, para minimizar o esforço do utilizador. Uma validação conceptual do otimizador demonstrou que a assistência otimizada resultou numa redução de 8.7% e 54.6% do CM e torque de interação do utilizador, respetivamente, comparativamente com a utilização do dispositivo em modo transparente.

O controlo HITL foi desenvolvido e integrado no exoesqueleto com sucesso. Os resultados de uma prova de conceito revelaram que este foi eficaz na redução do esforço do utilizador.

Palavras-chave: controlo *human-in-the-loop*, estimação do custo metabólico, exoesqueletos, lesões musculoesqueléticas relacionadas com o trabalho, trabalho assistido

Abstract

Work-related musculoskeletal disorders (WMSDs) affect roughly 60% of the European working population (1). Standing for long periods and carrying and lifting loads are all industrial activities that introduce significant loads on the workers' legs (1). Not every industry process can yet be fully automated, and, therefore, humans are still required to perform repetitive tasks in their workplaces (2). Exoskeletons can be employed for power augmentation of healthy workers by reducing the physical stress and strain on the user's muscles, however, today's exoskeletons can not yet perfectly synchronize to the human's intentions nor provide ideal and individualized assistance to each worker (2).

The main goal of this dissertation was to develop and validate a human-in-the-loop (HITL) control for a lower limb exoskeleton (LLE). This control aim was to optimize, in real-time, the torque profile of the exoskeleton, to a profile that minimizes the user's exertion, evaluated by the person's metabolic cost (MC) and interaction torque.

Firstly, a regression model capable of estimating, in real-time, a person's MC based on data from four inertial sensors was developed and integrated into the exoskeleton's system. The model was trained with data from three different activities: (i) standing; (ii) walking; and (iii) sitting. During a real-time experimental validation, the model achieved a root-mean-square error of 0.66 W/kg.

Then, a torque tracking control was developed for the LLE. This control ensured that the device's actuators track a pre-determined torque profile, generated by a natural cubic spline interpolator. It was composed of a mid-level controller that estimated the gait phase and generated a torque reference, and a low-level controller, used to drive the actuator according to the reference torque. A conceptual validation of the control showed that it was able to successfully assist the user by producing a comfortable gait.

Finally, the HITL control was developed, by integrating the MC estimating model and the torque tracking control with an optimization algorithm. This optimizer adapted the torque profile, namely the knee flexion and extension peak torque magnitudes, to minimize the user's exertion. A conceptual validation of the HITL strategy showed that the optimized assistance resulted in a reduction of 8.7% and 54.6% of one participant's MC and interaction torque, respectively, compared to using the device in a zero-torque mode.

The HITL control was successfully developed and integrated into an LLE. The proof-of-concept results revealed that this strategy was effective in reducing the user's exertion.

Keywords: assisted working, exoskeletons, human-in-the-loop control, metabolic cost estimation, work-related musculoskeletal disorder

Contents

- 1 Introduction 1**
 - 1.1 Motivation 1
 - 1.2 Problem statement 2
 - 1.3 Dissertation goals 3
 - 1.4 Research questions 4
 - 1.5 Contributions to knowledge 4
 - 1.6 Manuscript outline 5

- 2 Review on exoskeletons for assisting workers and human-in-the-loop controls 6**
 - 2.1 State of the art on industrial lower limb exoskeletons 6
 - 2.1.1 Introduction 6
 - 2.1.2 Methods 7
 - 2.1.3 Results 8
 - 2.1.4 Discussion 13
 - 2.2 State of the art on human-in-the-loop controls 16
 - 2.2.1 Introduction 16
 - 2.2.2 Methods 17
 - 2.2.3 Results 18
 - 2.2.4 Discussion 23
 - 2.3 State of the art on metabolic cost estimation models 26
 - 2.3.1 Introduction 26
 - 2.3.2 Methods 26
 - 2.3.3 Results 27
 - 2.3.4 Discussion 32
 - 2.4 General conclusions 33

- 3 System requirements and overview for assisted working 35**
 - 3.1 Industrial assistance requirements 35
 - 3.2 SmartOs system 37
 - 3.3 Proposed solution 39

- 4 Metabolic cost estimation 42**

4.1	Introduction	42
4.2	Methods	44
4.2.1	Dataset selection	44
4.2.2	Data preprocessing	47
4.2.3	Regression Models	49
4.2.4	Model integration in SmartOs' architecture	52
4.3	Validation	55
4.3.1	Offline	55
4.3.2	Bench tests	55
4.3.3	Human experiments	56
4.4	Results and discussion	62
4.4.1	Offline validation	62
4.4.2	Offline testing	68
4.4.3	Bench tests	69
4.4.4	Human experiments	70
4.4.5	Conclusions	77
5	Torque tracking control	78
5.1	Introduction	78
5.2	Methods	80
5.2.1	Torque profile generation	80
5.2.2	Torque controller implementation	82
5.3	Validation	84
5.3.1	Bench tests	84
5.3.2	Human experiments	85
5.4	Results and discussion	87
5.4.1	Bench tests	87
5.4.2	Human experiments	89
5.5	Conclusions	93
6	Human-in-the-loop control	94
6.1	Introduction	94
6.1.1	Covariance Matrix Adaptation – Evolutionary Strategy	95

6.2	Methods	98
6.2.1	Knee torque profile	98
6.2.2	CMA-ES optimizer	99
6.2.3	HITL control implementation	101
6.3	Validation	104
6.3.1	Bench tests	104
6.3.2	Human experiments	105
6.4	Results and discussion	107
6.4.1	Bench tests	107
6.4.2	Human experiments	109
6.5	Conclusions	115
7	Conclusions	116
7.1	Future work	119
	References	121
A	Appendix: Cross-correlation between Ingraham’s and InertiaLab’s 3D acceleration axes	131
B	Appendix: 3D acceleration acquired during the MC estimation validation	133

List of Figures

Figure 1 – LLEs for assisting industry workers.	8
Figure 2 – Description of each TRL.	10
Figure 3 – General control strategy of HITL algorithms.	17
Figure 4 – Industrial tasks more often associated with WMSDs.	35
Figure 5 – Knee module of SmartOs and respective components.	38
Figure 6 – Diagram of SmartOs architecture.	38
Figure 7 – InertiaLab system, where (a) presents the full InertiaLab’s architecture (84), and (b) and (c) present a single IMU with the lid closed and opened, respectively.	39
Figure 8 – Diagram of the proposed controller for optimizing the assistance of an active knee exoskeleton and minimizing the MC of a person in real-time.	40
Figure 9 – Woman wearing a respirometer device (K5, Cosmed, Italy).	43
Figure 10 – Representation of on-body location of the sensors of Ingraham’s dataset.	46
Figure 11 – Diagram of the process of segmentation of the dataset into 10-second windows, with and without the 5-second overlaps (bottom and top diagrams, respectively).	48
Figure 12 – Fluxogram depicting the code developed in SmartOs’ architecture that allowed for the MC estimation in real-time based on the data from 4 accelerometers.	53
Figure 13 – Representation of InertiaLab’s axis orientation in relation to Ingraham’s axis orientation.	54
Figure 14 – Interface options added to SmartOs’ APP that enabled the start of MC estimation.	54
Figure 15 – Locations of the lower body motion track markers.	57
Figure 16 – Participant instrumented with the wearable sensors, namely, the InertiaLab’s IMUS, motion track markers, HR chest monitor, and the respirometer device.	58
Figure 17 – Activities performed by the participants.	59
Figure 18 – Raw 3D acceleration signals measured by the InertiaLab’s IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 4.	60
Figure 19 – Heart rate of each participant during the experiment, and respective average.	61
Figure 20 – Diagram of the preprocessing steps that resulted in the best regression model performance.	63
Figure 21 – Layer diagram of the best performing CNN model.	66
Figure 22 – Comparison between the test subject’s true MC (orange) and the estimated MC (blue).	68
Figure 23 – Comparison between the test subject’s true MC (yellow), the estimated MC in MATLAB (blue), and the estimated MC in C++ (orange).	69

Figure 24 – Time diagram depicting the time it takes to execute the different MC estimation algorithm’s functions.	70
Figure 25 – Comparison between the MC estimated by the three methods of respirometry (per-breath, fast-estimated, and steady-state, in orange, black, and red, respectively) and the MC estimated by the EGPR model (in blue), for each participant.	71
Figure 26 – Boxplots presenting the MAPEs between the MC estimated by the EGPR model and the per-breath respirometry method. Each boxplot presents the results of the five participants (P1-P5) for a single activity.	74
Figure 27 – Bland-Altman plots depicting the agreement between the MC estimated by the EGPR model and the per-breath respirometry for each participant, where the black line represents the mean difference and the red and blue lines represent the 95% limits of agreement.	75
Figure 28 – Ankle torque profiles used on torque tracking controllers for a tethered exoskeleton (8) and an exoskeleton with electric actuators (93), and the natural ankle torque profile (95)	79
Figure 29 – Human knee torque profile when walking at different speeds (95).	80
Figure 30 – Curve shape of a torque profile with two flexion movements, with torque magnitude peaks at 20% and 60% of the gait cycle, and one extension movement with a torque magnitude peak at 87.5% of the gait cycle.	81
Figure 31 – Diagram of the torque controller integrated into SmartOs’ architecture.	82
Figure 32 – Interface option added to the SmartOs’ APP that enabled the start of torque control.	85
Figure 33 – Equipment worn by the participant during the torque controller’s validation.	86
Figure 34 – Tested torque profiles, for flexion and extension peak torques of 15 N.m and -15 N.m, respectively.	88
Figure 35 – Best performing knee torque profile.	89
Figure 36 – Actuator and reference torques during 30 seconds of torque control (colored blue, solid and dotted, respectively) and the PID command output (colored orange).	90
Figure 37 – Physiological and biomechanical variables measured during the human experimental protocol.	92
Figure 38 – Participant’s interaction torque and knee angle, measured during 30 seconds (between 3.5 and 4 minutes).	93
Figure 39 – Curve shape of the knee torque profile that was optimized by the HITL controller.	98

Figure 40 – Diagram depicting the different blocks of the HITL control strategy developed in SmartOs’ architecture.	101
Figure 41 – Code fluxogram of the CMA-ES optimizer developed in the LLOS board.	102
Figure 42 – Interface option added to SmartOs’ APP that enabled the start of HITL control and its relation to the previous adjustments.	104
Figure 43 – Equipment worn by the participant during the HITL controller’s validation.	106
Figure 44 – Time diagram depicting the time taken by the SmartOs system to execute the different HITL control algorithm’s functions.	108
Figure 45 – Flexion (positive) and Extension (negative) torque magnitude values for the peaks of the various torque profiles tested during optimization.	109
Figure 46 – Evolution of the CMA-ES objective function and respective variables.	110
Figure 47 – Optimized torque profile.	111
Figure 48 – Boxplots depicting the MC of the participant during five minutes of walking at 1.5 km/h without the exoskeleton, and with the exoskeleton in zero-torque mode and with the optimized torque controller.	112
Figure 49 – Interaction torque during five minutes (left) and 30 seconds (right) of walking at 1.5 km/h with a zero-torque controller (top) and the optimized torque controller (bottom). The red lines represent the maximum and minimum values of the interaction torque.	113
Figure 50 – Knee angle during five minutes (left) and 30 seconds (right) of walking at 1.5 km/h with a zero-torque controller (top) and the optimized torque controller (bottom).	114
Figure 51 – Raw 3D acceleration signals measured by the InertiaLab’s IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 1.	133
Figure 52 – Raw 3D acceleration signals measured by the InertiaLab’s IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 2 (Trial 1).	134
Figure 53 – 3D Acceleration signals measured by the InertiaLab’s IMUs that were not used to predict the MC (right waist, back waist, and right knee), for Participant 2 (Trial 1).	134
Figure 54 – Raw 3D acceleration signals measured by the InertiaLab’s IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 2 (Trial 2).	135
Figure 55 – 3D Acceleration signals measured by the InertiaLab’s IMUs that were not used to predict the MC (right waist, back waist, and right knee), for Participant 2 (Trial 2).	135
Figure 56 – Raw 3D acceleration signals measured by the InertiaLab’s IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 3.	136

Figure 57 – 3D Acceleration signals measured by the InertiaLab’s IMUs that were not used to predict the MC (right waist, back waist, and right knee), for Participant 3. 136

Figure 58 – Raw 3D acceleration signals measured by the InertiaLab’s IMUs that were used to predict the MC (chest, right wrist, left waist, and right ankle), for Participant 4. 137

Figure 59 – 3D Acceleration signals measured by the InertiaLab’s IMUs that were not used to predict the MC (right waist, back waist, and right knee), for Participant 4. 137

Figure 60 – Raw 3D acceleration signals measured by the InertiaLab’s IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 5. 138

Figure 61 – 3D Acceleration signals measured by the InertiaLab’s IMUs that were not used to predict the MC (right waist, back waist, and right knee), for Participant 5. 138

List of Tables

Table 1 – LLEs for power augmentation in the industry 9

Table 2 – "Chair-like" LLEs in the industry 11

Table 3 – Control strategies of the analyzed LLEs 12

Table 4 – Current major limitations of industrial exoskeletons 15

Table 5 – Studies conducted for the HITL optimization of the assistance of an exoskeleton, based on the user's MC 19

Table 6 – Studies conducted for the HITL optimization of the assistance of an exoskeleton, that used physiological signals other than the MC 21

Table 7 – Studies that proposed models for MC estimation using data acquired by wearable sensors 28

Table 8 – Industrial tasks more often associated with WMSD and their corresponding motor tasks, and the industrial sectors with a significant prevalence of those tasks 36

Table 9 – Human ROM of each joint during each of the relevant motor tasks, and the correspondent exoskeleton limits 36

Table 10 – Natural torque of each joint during each of the relevant motor task 37

Table 11 – Publicly available datasets used for estimating the MC 45

Table 12 – Conditions tested during the data preprocessing. Legend: ACC - "Acceleration"; BW - "Body weight" 49

Table 13 – Tested hyperparameters for the CNN 50

Table 14 – Machine learning models implemented in the Regression Learner App 51

Table 15 – Validation participants' demographics 57

Table 16 – Performance comparison between the model that was trained with the HR, acceleration, and BMI, and the model that was trained with the acceleration and BMI 64

Table 17 – Validation and test performance of the three activity-specific models (walking, sitting, and standing) and the general model 65

Table 18 – Validation RMSE and R^2 of each regression model trained 67

Table 19 – Validation RMSE of an EGPR model with a Sigma parameter optimized by different methods (BO, grid search, and random search) in comparison to no optimization process 68

Table 20 – RMSE between the MC estimated by the EGPR model and the MC estimated by each respirometry method (in W/kg), for each activity (standing, walking at 1.5 km/h, 2km/h, and 3 km/h, and sitting), and for each participant (P1-P5). The average RMSE for each activity is presented in the last row 73

Table 21 – Average delay between the estimation made by the regression model and the respirometer, and RMSE of the time agreement between the two methods, for each participant (P1-P5)	76
Table 22 – Best PID gains values	87
Table 23 – Initial parameters' values for the CMA-ES optimizer	100
Table 24 – Values of the parameters used for the termination conditions of the CMA-ES optimizer.	101
Table 25 – Best objective function weights	108
Table 26 – Wrist axis cross-correlation between Ingraham's and InertiaLab's data. Legend: IL - "InertiaLab"	131
Table 27 – Chest axis cross-correlation between Ingraham's and InertiaLab's data. Legend: IL - "InertiaLab"	131
Table 28 – Waist axis cross-correlation between Ingraham's and InertiaLab's data. Legend: IL - "InertiaLab"	131
Table 29 – Ankle axis cross-correlation between Ingraham's and InertiaLab's data. Legend: IL - "InertiaLab"	132

Acronyms

APP Application (Software)	IDE Integrated development environment
BDTR Boosted decision tree	IMU Inertial measurement unit
BMI Body mass index	KPI Key performance indicator
BO Bayesian optimization	LLE Lower limb exoskeleton
CCU Central controller unit	LLOS Low-level orthotic system
CNN Convolutional neural network	LLOCV Leave-one-out cross-validation
CMA-ES Covariance matrix adaptation evolution strategy	LR Linear regressor
DOF Degree of freedom	MC Metabolic cost
ECG Electrocardiogram	MAD Mean absolute deviation
EGPR Exponential gaussian process regressor	MAPE Mean absolute percentage error
EMG Electromyography	NN Neural network
FDA Food and drug administration	PID Proportional-integral-derivative
GPR Gaussian process regressor	ReLU Rectified Linear Unit
GRF Ground reaction force	RMSE Root-mean-square error
GSR Galvanic skin response	ROM Range of motion
GSVM Gaussian support vector machine	SVM Support vector machine
HITL Human-in-the-loop	TRL Technology readiness level
HR Heart rate	WMSD Work-related musculoskeletal disorder
	WMSS Wearable motion system

1 Introduction

This manuscript presents the work developed for this dissertation, entitled "Human-in-the-loop control for lower limb assistance driven by exoskeleton", in the scope of the fifth year of the Integrated Master's in Biomedical Engineering during the academic year of 2022/23, in the field of Medical Electronics, at the University of Minho. The work presented in this dissertation was developed at the Biomedical Robotic Devices Laboratory (BiRD Lab), of the Centre of MicroElectroMechanical Systems (CMEMS) Research Centre established at the University of Minho. This work is immersed in a research project that resulted from the collaboration of the University of Minho and 'Bosch Car Service' (Braga, Portugal) named "Connected Manufacturing – Digital Transformation".

During this period, it was developed a machine learning model capable of estimating a person's metabolic cost (MC) in real-time. This model was then integrated into a human-in-the-loop (HITL) control, of a lower limb exoskeleton (LLE), capable of optimizing a torque controller to minimize the user's exertion in real-time. The followed methods during the dissertation, its results, and the conclusions taken are presented in this manuscript.

1.1 Motivation

An ongoing cause of concern in the industry are work-related musculoskeletal disorders (WMSDs), the most common work-related health problem in developed countries, affecting millions of European workers across all sectors and occupations (1). In total, it is estimated that 1.71 billion people worldwide have a musculoskeletal disorder (3). Musculoskeletal disorders are injuries that limit human motion by affecting muscles, nerves, tendons, ligaments, and so on. Some examples of such disorders are carpal tunnel syndrome, tendonitis, tendon strains, and ligament sprains. A report commissioned by the European Agency for Safety and Health at Work, in 2019, disclosed that three out of five workers in the European Union have musculoskeletal complaints, and more than half of the workers with musculoskeletal disorders reported being absent from work, at least once, due to the illness (1).

The prevalence of WMSDs in industry workers is around 29%, 11%, 33%, and 17% for the leg, hip, knee, and ankle/feet, respectively (4). In 2020, body movements under/with physical stress accounted for 18.4% of the causes of non-fatal work accidents (5). The industry activities that are more critical for WMSDs in the lower limbs are standing for long periods and carrying/lifting heavy loads since the three primary risk factors comprise task repetitions, forceful exertions, and awkward postures (6). Pain in the lower limbs was a usual (49.5%) complaint among hairdressers (7), workers required to stand for long times, and 63%

of workers that frequently carry and/or move heavy loads reported having musculoskeletal disorders (1). The industry sectors with a higher prevalence of lower limb WMSDs, in 2015, were agriculture, forestry, and fishing (46%), construction (40%), and water supply (40%) (1).

Over the past years, various ergonomic studies have proposed strategies to reduce the physical load of workers in their workplace, such as resting periods, task rotations, and the automation of industry tasks. However, not every process can be fully automated, and humans are still required to deal with significant loads, repetitive tasks, and non-ideal postures, e.g., in transporting materials on rough terrains (2).

LLEs can be employed for power augmentation of healthy workers by reducing the physical stress and strain on the users' muscles as they perform motions like walking, squatting, stand-to-sit, and sit-to-stand, or while they are in stationary positions (either sitting or standing) (2).

1.2 Problem statement

Despite the advantages of using LLEs for human assistance and power augmentation, various limitations to these devices can still be identified. These limitations are the reason for the lack of exoskeletons being developed, commercialized, and used in industrial applications. One major challenge of exoskeletons is the need for compliant, compatible, and safe human-robot interactions. However, today's exoskeletons can not yet provide ideal and individualized assistance to each user (2, 8).

The synchronization between the human's and robot's motions is still a complex problem, as movement patterns can diverge significantly between different people. Typically, exoskeletons' control parameters (joint positions, torque profiles, etc.) are chosen based on the biomechanics of the average population (9, 10) and are generally good enough to assist their users, but they do not take any physiological parameter into account, like pain, effort, or metabolic consumption, nor coordinate the device according to the human's intentions (11).

This raises the need for more intelligent control schemes, that can adapt the exoskeleton assistance to the human's needs by monitoring the synergy between the user and the device. Several controllers have recently been proposed by taking inspiration from the human's gait, which is smooth, comfortable, and naturally optimized for each specific person (8).

One possible strategy is the real-time optimization of the control parameters based on physiological signals obtained from the user – HITL control (12). This way, by monitoring data acquired by wearable sensors, it is possible to continuously adapt the assistance provided by the device to each user, and humans are actively involved in the exoskeleton control. This strategy shows a huge potential for industrial

applications, as it could be used to minimize the MC of workers performing heavy tasks. However, it has yet to be implemented for industrial exoskeletons assisting workers in real-time (13).

The physiological parameter most optimized in these approaches is the MC. However, the standard method for the estimation of this signal is through indirect calorimetry, which requires expensive and not portable equipment, is time-consuming, its estimation is noisy, and is not feasible for real-world applications (14). To overcome this drawback, various machine and/or deep learning models have been proposed to estimate the MC based on data obtained from portable sensors (15, 16).

1.3 Dissertation goals

The main goal of this dissertation is the development and preliminary validation of an adaptive torque control for an LLE towards the power augmentation of workers during physically demanding tasks for the lower limbs, such as standing for long periods and carrying/lifting heavy loads. This control will implement a HITL algorithm to optimize the interaction between humans and an LLE, with a focus on the knee joint. The HITL algorithm aims for the real-time optimization of an exoskeleton's knee joint torque by minimizing the user's exertion, evaluated by two parameters: (i) the user's MC, estimated in real-time by a regression algorithm based on non-intrusive wearable sensor data, and (ii) the interaction torque between the user and the device.

To reach the proposed main goal, four smaller objectives were identified:

Objective 1: Perform a literature review of LLEs, HITL control strategies, and regression algorithms for MC estimation. More precisely, survey the literature on LLEs developed for assisting industry workers, on the existing HITL controls, and on MC estimation models proposed for clinical or daily assistance. The key performance indicator (KPI) for this objective is the production of the review. This objective is presented in Chapter 2.

Objective 2: Develop a regression model for MC estimation based on data obtained from wearable and non-intrusive sensors, in real-time. This objective includes: (i) the search for a dataset with relevant sensor data collected during motor tasks similar to standing, walking for long periods, and carrying/lifting; (ii) the selection of minimal sensor inputs toward maximum usability in the industry; (iii) the development and benchmark of several regression models for MC estimation; and (iv) the implementation of the regression model into the LLE's architecture. As a KPI, the regression model should obtain a root-mean-square error (RMSE) below 0.8 W/kg and a coefficient of determination larger than 80% (17, 18). This objective is addressed in Chapter 4.

Objective 3: Develop a torque tracking control that manipulates an LLE's knee joint to follow

a desired torque trajectory. This objective includes the development of: (i) a natural cubic interpolator for generating the torque profile; (ii) a mid-level controller capable of estimating the gait cycle phase and generating the reference torque from the torque profile; and (iii) a low-level controller that minimizes the difference between the reference and the actuator's torque. Guaranteeing the user's comfort level when walking with the controller is the KPI for this objective. This objective is presented in Chapter 5.

Objective 4: Develop a HITL adaptive torque control capable of adjusting the LLE's knee joint torque to minimize the exertion level of the users, estimated by the regressor model. This adaptive torque control should be able to find a customized torque profile for each user according to his/her exertion level, measured by the weighted sum of the MC and the human-exoskeleton interaction torque. The KPI is the development of a control that requires less than 30 minutes of optimization (8, 19). This objective is presented in Chapter 6.

Objective 5: Validate the regression model and the proposed controls during human tests walking under the assistance of the LLE. The validation comprises the development of the experimental protocols, the data acquisition, and the evaluation of the regression model's and controllers' performance. The HITL control should grant a reduction of the MC and the interaction torque of the exoskeleton's users by at least 10% (8, 19), compared to performing the same activities with the device in zero-torque mode. This objective is presented in Chapters 4 and 6.

1.4 Research questions

This proposed dissertation was developed in order to find the answer to the following research questions (RQs):

RQ1: What are the current effects of HITL controls implemented on LLEs? This RQ is related to Objective 1 and is answered in Chapter 2.

RQ2: How accurate can a regression model be in estimating the MC based on a small number of wearable and non-intrusive sensors? This RQ is related to Objective 2 and is answered in Chapter 4.

RQ3: How much can LLE users' exertion be reduced by a real-time HITL optimization? This RQ is related to Objective 5 and is answered in Chapter 6.

1.5 Contributions to knowledge

The main contributions to knowledge achieved by this dissertation are:

- A review of the LLEs for power augmentation and the "chair-like" exoskeletons developed for industrial applications (workers' assistance).
- A review of HITL control strategies used to optimize the assistance of an LLE regarding the users' needs.
- A review of machine and deep learning models used to estimate the MC, in real-time, based on wearable sensors.
- A novel regression model used to estimate the MC based on a minimal number of acceleration sensors during various activities (walking, standing, and sitting), possible to be worn simultaneously with an LLE, and feasible for HITL applications in assisted working.
- A torque tracking control that enabled the manipulation of an exoskeleton's knee joint according to a pre-determined torque profile, which was tuned to guarantee the user's comfort when walking.
- A novel HITL control capable of finding customized optimal assistance that minimizes the MC and interaction torque of an LLE user, and differentiates itself from the literature's studies by minimizing an MC estimated in real-time with a regression model.

The work developed in this dissertation also resulted in the publication of the following conference paper:

- Monteiro, S., Figueiredo, J., Santos, C. "Towards a more efficient human-exoskeleton assistance", IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Tomar, 2023.

1.6 Manuscript outline

This document is organized into six additional chapters. Chapter 2 comprises the reviews of industrial LLEs, HITL controls for exoskeletons, and MC estimation models. Chapter 3 presents the description of the systems used in this dissertation and the requirements set for using LLEs for assisted working. In Chapter 4, the development of the regression model used to estimate the MC is described, and its validation during human tests is presented. Chapter 5 presents the development of the torque-tracking control and its validation as a proof-of-concept during human experiments. Chapter 6 presents the HITL control developed in a knee exoskeleton, based on the achievements made in Chapters 4 and 5, and its validation in preliminary human tests. Chapter 7 presents the conclusions made in the dissertation and future work.

2 Review on exoskeletons for assisting workers and human-in-the-loop controls

In this chapter, the literature review on the dissertation topics is presented. Firstly, the LLEs used for industrial applications were surveyed. Then, this chapter presents the analysis of LLEs' HITL controllers capable of real-time optimization of the users' MC. Finally, the review on regression models used for MC estimation based on data acquired by wearable sensors is also presented. Each literature review is complemented by a critical analysis of the state of the art. This chapter ends with a synthesis of the conclusions taken from these reviews.

2.1 State of the art on industrial lower limb exoskeletons

2.1.1 Introduction

Despite the increase in automatization and mechanization that the industry is currently experiencing, a lot of workers are still daily exposed to physical workloads and WMSDs are still a significant issue that employees must face (20). Unfortunately, strategies to avoid physically demanding occupational tasks, like material handling, repetitive movements, and awkward postures can not always be implemented because of economic reasons or certain workplace characteristics. An alternate direction must be followed to parry this complication, one that can free workers from the burden of tough manual work, lessen the likelihood of injury, and improve work efficiency: the use of exoskeletons (2, 21).

An exoskeleton is a wearable mechanical structure that fits closely to the user's body and is operated to assist the wearer according to their needs thanks to actuators that reduce the loads on the human joints. These devices should not be confused with orthosis, which are robotic accessories that enable a disabled person to move more naturally, whereas an exoskeleton is better described as a structure that augments the performance of its wearer (22).

Exoskeletons can be classified according to: (i) the body parts they support as lower-body, upper-body, or full-body; (ii) the actuation they provide to the human joints as active, passive, or quasi-passive; and (iii) the level of resemblance they have to a human as anthropomorphic, non-anthropomorphic or quasi-anthropomorphic (20).

Exoskeletons' effects on assisted working are still not deeply known due to a lack of studies. The few studies that can be found seem to back up the idea that the use of such devices leads to a decrease in

workers' physical stress (joint loads) and strain (muscle activity, discomfort, and fatigue), but the general perceived strain felt by the users seem to increase, probably because of the discomfort associated with the operation of an exoskeleton (21). So, it seems that, by providing an external torque to the joints of the user, an exoskeleton can reduce the wearer's muscle activity and support the addressed musculoskeletal structures. This leads to an increase in human strength and endurance and allows workers to perform certain tasks that otherwise could not be independently achievable (2, 21). The objective of this section is to provide an overview of industrial LLEs that are already being used to increase the strength of workers in occupational tasks and devices that are currently under development.

2.1.2 Methods

Search methodology

The literature search was conducted in the online database Scopus with the following combination of keywords: exoskeleton AND (("lower limb" AND industry) OR (bleex) OR ("Hanyang Exoskeleton Assistive Robot" OR "HEXAR")) OR "legx" OR (load AND lifting AND assistance AND "lower limb"). Additionally, the search was extended to the websites of the commercialized exoskeletons whose descriptions could not be found in any of the papers. The search was executed between the 21st of September of 2022 and the 30th of August of 2023, and no restriction related to the papers' release date was taken into account.

Selection strategy

The papers were selected based on the following criteria: (i) presented an LLE used for industrial applications; (ii) presented a detailed description of the device's hardware and control architecture. Passive devices were not excluded from the search. Additionally, only one paper was selected for each exoskeleton, namely the paper with a higher level of detail regarding the technical features of the device.

Data extraction

The selected papers were analyzed in order to obtain the following information: (i) the device presented; (ii) the motor and industrial tasks assisted by the device; (iii) the weight of the exoskeleton and its maximum payloads; (iv) the number of joints and degrees of freedom (DOFs); (v) the torque provided to the users; (vi) the range of motion (ROM) of each joint; (vii) the locomotion speeds allowed; (viii) the sensors implemented in the device; (ix) the controller architecture, and (x) and the device's limitation. The extracted information served as the benchmark for the discussion of the current state of industrial LLEs.

2.1.3 Results

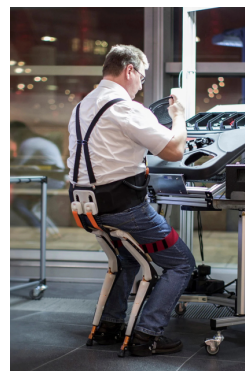
The search methodology resulted in 94 papers, that were filtered, following the mentioned selection strategy, and reduced to 12 articles. The literature available revealed that the function and areas supported can vary a lot across the LLEs commercially available, but two main types were identified: some assist the human joints in movements like walking, squatting, or kneeling, while the wearer carries significant loads – exoskeletons for power augmentation; others provide support for prolonged standing or sitting positions, while the worker performs repetitive and stationary assembly tasks, with their upper limbs – “chair-like” exoskeletons (21). Figure 1 presents some of the LLEs developed for and used in industrial applications, where Figures 1a and 1b depict two devices used for power augmentation, while Figures 1c and 1d depict two chair-like exoskeletons.



(a) BLEEX. Taken from: (23)



(b) Hercules. Taken from: (24)



(c) Chairless Chair. Taken from: (25)



(d) legX. Taken from: (26)

Figure 1: LLEs for assisting industry workers.

Exoskeletons for power augmentation

Many work environments do not allow the transportation of heavy objects by wheeled vehicles, like rough slopes and staircases, making legged locomotion the desired method of transportation. LLEs can help an able-bodied person carry heavy loads, in those harsh locations, and even assist them in performing the weight-lifting movement itself (2, 27).

An overview of power-augmentation LLEs can be found in Table 1. There, it is possible to analyze the assessed technology readiness level (TRL), the motor and industrial tasks assisted, the weight and load capacity, the joints, the actuated DOFs in the sagittal plane and the type of actuator used, the torque applied to the joints, the ROM of each joint and the sensors implemented in each LLE. The BLEEX and Hercules devices are presented in Figure 1a and 1b, respectively.

Table 1: LLEs for power augmentation in the industry. Legend: abd/add - "Abduction/Adduction"; AJ - "Ankle joint"; dors/plant - "Dorsiflexion/Plantarflexion"; flex/ext - "Flexion/Extension"; GRF - "Ground reaction force"; HJ - "Hip joint"; KJ - "Knee Joint"; N/A - "Not available/applicable"; SEA - "Series elastic actuator"

Exoskeleton	TRL	Motor task assisted	Industrial task	Weight	Max Payloads	Joints per leg	Active actuators	Type of active actuators	Torque	ROM (Sag. plane)	Sensors
RoboKnee (28)	4	Walking, climbing stairs and kneeling	Carrying heavy loads	<3 kg	60 kg	KJ (1 DOF)	KJ: flex/ext	SEA	N/A	N/A	Linear encoder (knee), GRF foot sensors
NTU's (29)	4	Walking	Carrying heavy loads	34,6 kg	20 kg	TJ, HJ, KJ, AJ (6 DOF)	TJ: flex/ext HJ: flex/ext KJ: flex/ext AJ: dors/plant	Electric motor	N/A	HJ: [-20,60] KJ: [0,120] AJ: [-30,20]	Rotary encoder (knee), GRF foot sensors
BLEEX (9)	7	Walking (4,68 km/h)	Carrying heavy loads	27 kg	50 kg	HJ, KJ, AJ (7 DOF)	HJ: flex/ext HJ: abd/add KJ: flex/ext AJ: dors/plant	Hydraulic actuators	N/A	HJ: [-45, 45] KJ: [0, 121] AJ: [10, 121]	Encoders, linear accelerometers, force sensors, load distribution sensors, inclinometers
HULC (30)	9	Walking (11 km/h)	Carrying heavy loads	24 kg*	91 kg	HJ, KJ, AJ (7 DOF)	HJ: flex/ext HJ: abd/add KJ: flex/ext AJ: dors/plant	Hydraulic actuators	N/A	N/A	N/A
HEXAR (31)	7	Walking (3 km/h)	Carrying heavy loads	23 kg	30 kg	HJ, KJ, AJ (7 DOF)	HJ: flex/ext KJ: flex/ext	Electric motor	34 Nm (continuous) 57 Nm (stall)	HJ: [-40, 150] KJ: [0, 127] AJ: [-30, 30]	GRF foot sensors, torque sensors, force/torque sensor (harness)
Hercules (32)(33)(34)	9	Walking (5 km/h)	Carrying heavy loads	30 kg	40 kg	HJ, KJ, AJ (7 DOF)	HJ: flex/ext KJ: flex/ext	Electric motor	N/A	N/A	N/A
Sado's (35)	6	Walking and squatting	Carrying and lifting heavy loads	N/A	N/A	HJ, KJ, AJ (6 DOF)	HJ: flex/ext KJ: flex/ext	SEA	HJ: 38 Nm KJ: 17 Nm	Gait / Squat HJ: [0,30] / [0,80] KJ: [0,38] / [0,80] AJ: [-21,35] / N/A	GRF foot sensors, joint angle potentiometers (hip/knee)
MIT's (10)	6	Walking (3,6 km/h)	Carrying heavy loads	11,7 kg	>36 kg	HJ, KJ, AJ (4 DOF)	None	N/A	N/A	N/A	Strain gauges (lower leg), rotary potentiometer (knee)
Capitani's (27)	3	Standing and squatting	Shotcrete projection and lifting heavy loads	6 kg	N/A	HJ, KJ, AJ (6 DOF)	None	N/A	N/A	N/A	N/A

* without batteries

The TRL was used to analyze the state of development of each exoskeleton, useful for comparison purposes. Since no information on this matter could be found in the literature, the values provided were self-accessed according to the descriptions of the levels presented in Figure 2.

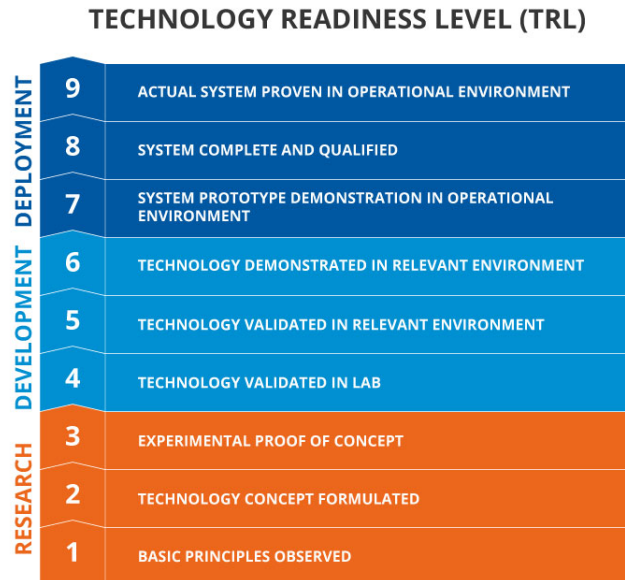


Figure 2: Description of each TRL. Taken from: (36).

”Chair-like” exoskeletons

“Chair-like” exoskeletons are wearable lightweight passive devices that support their users’ weight when they want to stand or sit for long periods, anytime and anywhere, by “locking” the knee joint. These ergonomic exoskeletons do not constrain any type of primary movement and allow the worker to freely walk when performing other tasks, while still wearing the devices.

An overview of the “chair-like” exoskeletons can be found in Table 2, where the TRL, the motor and industrial tasks assisted, the weight, the joints per leg, the sensors, and the special features of each device were summarized. The Chairless Chair and legX devices are presented in Figure 1c and 1d, respectively.

Table 2: "Chair-like" LLEs in the industry. Legend: AJ - "Ankle joint"; GRF - "Ground reaction force"; KJ - "Knee joint"; N/A - "Not available/applicable"

Exoskeleton	TRL	Motor task assisted	Industrial task	Weight	Joints per leg	Sensors	Height settings
Chairless Chair (37)	9	Standing and sitting	Stationary assembly tasks	2,95 kg	KJ (1 DOF)	N/A	Manually adjustable heights settings: standing, high (125° knee angle), and low sitting (90° knee angle)
ArcheleisFX (38)	9	Standing and sitting	Stationary assembly tasks	4,2 kg	KJ (1 DOF)	N/A	Two possible height settings: standing with the knees slightly bent and normal standing
legX (39)	9	Sitting and squatting	Stationary assembly tasks and lifting loads	6 kg	KJ (1 DOF)	N/A	One possible height setting (110° knee angle)
ChairX (40)	6	Sitting	Stationary assembly tasks	13.4 kg	KJ, AJ (1 DOF)	Joint angle sensors	Possible height settings at every 10° increment (10°, 20°, ..., 360°)
E-LEG (41)	6	Sitting and squatting	Stationary assembly tasks	3 kg	KJ (1 DOF)	Inertial sensor (knee angle), GRF foot sensors	Multiple height settings available
KARE-1 (42)	6	Sitting and squatting	Stationary assembly tasks	6 kg	KJ (1 DOF)	Angular velocity sensors, GRF foot sensors	Fixed sitting height
CEX (43)	9	Sitting	Stationary assembly tasks	1.6 kg	KJ (1 DOF)	N/A	Two possible height settings (55°, 70°, and 85°)

Control strategies of LLEs used in industrial applications

Control loops can be essentially divided into three layers: high-, mid-, and low-level. The high-level block receives information from the environment and/or the user to detect changes in the surroundings and/or the user's intentions. The mid-level controller is used to continuously compute the joint's desired torque or position and is typically divided into two sub-layers: The detection/synchronization block that outputs the gait phase (0-100%) or the gait state (swing, stance, etc.); The action block that outputs a motor command that controls the kinematics (angle/speed) or the kinetics (forces/torques) of the LLE. The final layer is the low-level controller used to manipulate the position or the torque of the actuators (44).

Baud et al. (44) studied the control strategies of 291 LLEs, identifying the most common blocks used in the device's control loops and describing the different possible implementations. Based on the nomenclature used by Baud et al., the control strategies of the LLEs presented in the previous sections are shown in Table 3, where the high-, mid-, and low-level blocks are presented.

Table 3: Control strategies of the analyzed LLEs. Based on: (44). Legend: N/A - "Not available/applicable"

Exoskeleton	High-level control	Mid-level control		Low-level control
		Detection	Action	
Roboknee	N/A	Gait phase detection	Function of joint instant state	Closed-loop torque controller
NTU's	N/A	Gait phase detection	Position profile	Position controller
BLEEX	N/A	Gait phase detection	Joint torque estimation (Method unknown)	Position controller (stance) Closed-loop torque controller (swing)
HULC	N/A	N/A	N/A	N/A
HEXAR	N/A	Gait phase detection	Joint torque estimation (Torque sensor)	Closed-loop torque controller
Hercules	N/A	N/A	N/A	N/A
SADO's	Movement recognition	State-machine function	Joint torque estimation (Dual extended Kalman filter)	Gait-cycle dependent torque controller
MIT's	N/A	State-machine function	Position profile	Gait-cycle dependent damper controller
Capitani's	N/A	N/A	N/A	N/A
CC	Manual user input	N/A	Body weight support	Fully Passive
Archelis	N/A	N/A	Body weight support	Fully Passive
legX	Manual user input	Event trigger	Body weight support	Fully Passive
ChairX	Movement recognition Manual user input	State-machine function Event trigger	Position profile Body weight support	Position controller
E-LEG	N/A	State-machine function	Body weight support	Fully Passive
KARE-1	Manual user input	N/A	Body weight support	Position controller
CEX	N/A	N/A	Body weight support	Fully Passive

2.1.4 Discussion

From the literature review provided in Table 1, it was possible to conclude that exoskeletons for power augmentation are mainly used for assisting users in load carrying. It also showed that most of the devices were actively actuated in, at least, one joint and that most of these LLEs were constituted by electric motors (29, 31, 32). Regarding the torque provided by the active actuators, barely any information could be found, except on two papers that provided 34 Nm/38 Nm (31, 35) and 17 Nm (35) of continuous torque in the hip and knee joint, respectively. The joint that was more commonly actively actuated was the knee, followed by the hip, both for walking and lifting assistance, with an emphasis on motions on the sagittal plane. In terms of the joints' ROM, the devices analyzed differed significantly from each other but in general, the hip, knee, and ankle joints' ROM ranged between -40° to 150° , 0° to 127° , and -30° to 121° , respectively (9, 29, 31, 35).

Additionally, it was found that "chair-like" exoskeletons are ideal for the assistance of workers performing tasks in stationary positions, that do not move for long periods of time, like the ones presented in Table 2. Most of these devices focus on knee assistance by locking this joint and allowing the person to sit or stand anywhere, making those tasks less strenuous on the leg muscles, representing, therefore, passive devices without active actuation (37–39, 41, 43). Additionally, two of the "chair-like" exoskeletons presented were semi-active devices, as one was composed of an active leg support unit (40), and the other of an active rotary actuator (42). The ROM of the knee joint of such devices varied depending on the allowed sitting height, but little information was found regarding this requirement.

The weight of the devices varied significantly depending on the number of actuated joints and the type of actuation. Fully passive exoskeletons with one-joint assistance (the "chair-like" exoskeletons) weighted between 1.6 kg to 6 kg (37–39, 41, 43) while passive exoskeletons with three-joint assistance weighted from 6 kg to 11,7 kg (10, 27). Actively actuated LLEs with three-joint assistance weighted significantly more, from 23 kg to 34,6 kg (9, 29–32). This showed that a trade-off has to be made between the weight of the device and the torque provided by the joints (level of assistance). Additionally, the maximum payload of the devices was also limited, varying from 20 kg (29) to 91 kg (30), and was generally directly correlated with the LLE weight.

In terms of the type of sensors normally used in the LLEs, the most common were encoders/potentiometers used to measure joint angles (9, 10, 28, 29, 35, 40, 41) and ground reaction force (GRF) sensors used to measure the force between each foot and the floor (10, 28, 29, 31, 35, 39, 42). Other sensors were also used, depending on the type and features of each LLE controller, such as

accelerometers (9) to measure the joints' acceleration, human-robot force sensors (9) and strain gauges (10) to measure the interaction force between the human and the robot, load distribution sensors (9) to measure the center of pressure of the user, inclinometers (9) to measure the orientation relative to gravity, torque sensors (31) to measure the torque applied by an active actuator, and absolute encoders (42) to measure the angular velocity of the joints.

Regarding the control strategies followed by the analyzed devices, presented in Table 3, there was a significant difference between the devices used for power augmentation (mainly active) and the "chair-like" exoskeletons (mainly passive) since the passive devices required no torque nor damping control. Table 3 also showed that most of the exoskeletons used for power augmentation did not have any high-level control, except the exoskeleton that was used for both carrying and walking loads (35), that used movement recognition to detect the humans' intentions. In fact, recognizing human intentions is essential for industrial exoskeletons that need to be employed in spaces with multiple terrain types (e.g. stairs and ramps) and are worn by workers who are in constant movement and need to perform various tasks.

Additionally, there is a lack of controllers that enable smooth and imperceptible transitions between assistance modes. Furthermore, the exoskeletons' controllers were designed to assist the average person and did not take any physiological parameter into account, such as their effort or MC.

Limitations of industrial exoskeletons

Some concerning limitations were found across the literature, which varied depending on the industrial task being performed and the type of actuators incorporated in the devices (21). In Table 4, the most relevant limitations to the implementation of LLEs by companies are presented, for each of the industry applications discussed and actuation type (2, 20, 21, 45).

Additionally, some general limitations, independent of the exoskeleton type and application, were also identified in the literature. Namely, the lack of universal safety standards for industrial applications of exoskeletons imposes an important barrier to their adoption by companies. Secondly, the impossibility of an exoskeleton to be worn by everyone (due to weight and anatomic limitations) and the general increase in MC and discomfort associated with wearing LLEs are restraining their acceptance by the workers (2, 20). Furthermore, the exoskeletons' controllers focus on either fixed trajectories or trajectories adapted to the gait phase, and not on adapting their assistance to the workers' exertion.

Table 4: Current major limitations of industrial exoskeletons (2, 20, 21, 45)

Industrial task	Actuator type	Limitations	
Lifting and carrying heavy loads	Active	<ul style="list-style-type: none"> - Heavy, big, and noisy devices; - Short battery span; - User's movements are limited to walking in a rigid gait; - More expensive; - Mechanisms that automatically recognize human intentions to move are still limited; - The limited technology state of actuators and power supply sources; 	<ul style="list-style-type: none"> - The design of exoskeletons is a complex process due to the kinematic complexity and variability of the muscle-skeletal system; - Lack of studies that analyze the efficacy and effects of wearing exoskeletons; - Assistance limited to specific motor tasks; - Donning and doffing takes too long and requires assistance; - User's speed is very limited;
	Passive	<ul style="list-style-type: none"> - Can't fully assist the joints with the necessary torque; 	
Stationary assembly tasks	Passive	<ul style="list-style-type: none"> - Sitting height is normally non-adaptive; - Movements such as ascending and descending stairs can be more difficult to perform, and some others can even be completely restricted, like kneeling; - Can lead to discomfort during long-term sitting; - Needs for repetitive donning and doffing are time-consuming; 	

Risks of LLEs

Despite all the benefits that LLEs provide to their users, they are also exposed to certain risks that subject their health to danger. These risks are often ignored by the literature, and rarely candidly discussed, hindering the evaluation of risk-to-benefit ratios of exoskeletons.

In 2013, on an evaluation review of the ReWalk - a rehabilitation exoskeleton - written by the U.S.A. Food and Drug Administration (FDA), several risks were identified (46). Despite being associated with that specific device, the exposed dangers could also be extended to other LLEs because of the shared design principles between devices. The possible adverse events that could occur when using an LLE identified by the FDA were: (i) instability, falls, and associated injuries; (ii) Bruising, skin abrasion, pressure sores, and soft tissue injuries; (iii) hypertension, and changes in blood pressure and heart rate (HR); (iv) adverse tissue reactions; (v) electrical interference with other devices; (vi) burns and electrical shocks; (vii) device malfunctions; and (viii) use errors (46).

In 2017, He et al. (47) accessed the reports of adverse events involving several rehabilitation exoskeletons. In this study, two other risks of wearing LLEs were identified: Bone fractures not caused by falls and long-term secondary effects. More recently, Rodríguez-Fernández et al. (48) studied the adverse events experienced by patients with muscular impairments while wearing an LLE. Out of the 87 studies reviewed, of a list of 25 different devices, 36 studies reported the existence of adverse events. Besides the risks recognized by the FDA, other secondary effects were reported: fatigue of the upper

limbs, low back pain, urinary infections, and dizziness (48).

In general, it was possible to conclude that LLEs are still barely used in industry and commercialized devices focus on "chair-like" assistance, instead of power augmentation, due to the limitations and risks of the heavier, but powered, devices. HITL controllers that focus on the optimization of the assistance to each user, in an individualized approach, could diminish the challenges of the LLEs and increase the users' acceptance of the devices.

2.2 State of the art on human-in-the-loop controls

2.2.1 Introduction

Various studies have shown that the energy expended by humans during various tasks can be reduced by wearing LLEs. However, the reduction verified could, potentially, be maximized by implementing user-oriented assistive strategies.

Typical assistance strategies can only evaluate the performance of exoskeletons and change their design if needed after they are effectively used and physiological data is obtained - offline optimization. Additionally, natural differences between users often result in inter-subject variability in their responses to the same device with the same control strategy (11, 13).

HITL algorithms can be implemented in LLEs for the automatic tuning of desired controller parameters (like the assistive torque) depending on real-time physiological measurements (like energy consumption). These reinforcement learning methods operate on "trial-and-error" approaches until finding the optimal assistance profile for each user – online optimization (13). Recently, various studies showed that this alternative control method allowed for improved responses to powered devices due to the individualized approach of the assistance, which is automatically tailored for each specific user. Figure 3 depicts the general control loop of a HITL strategy.

HITL control strategies for LLEs normally focus on the minimization of a cost function based on the MC of the user. In order to estimate the MC, indirect calorimetry methods are normally applied, by using respiratory measurements. This method utilizes the Brockway equation in order to convert carbon dioxide (CO_2) and oxygen (O_2) rates [mL/sec] into energy consumption [W]. Paired with the Brockway equation, a first-order dynamic model is usually used in HITL applications to obtain the instantaneous energetic consumption for faster estimation, since the indirect calorimetry method can only provide an accurate measurement after the MC steady-state is achieved (roughly two to three minutes after starting a new activity).

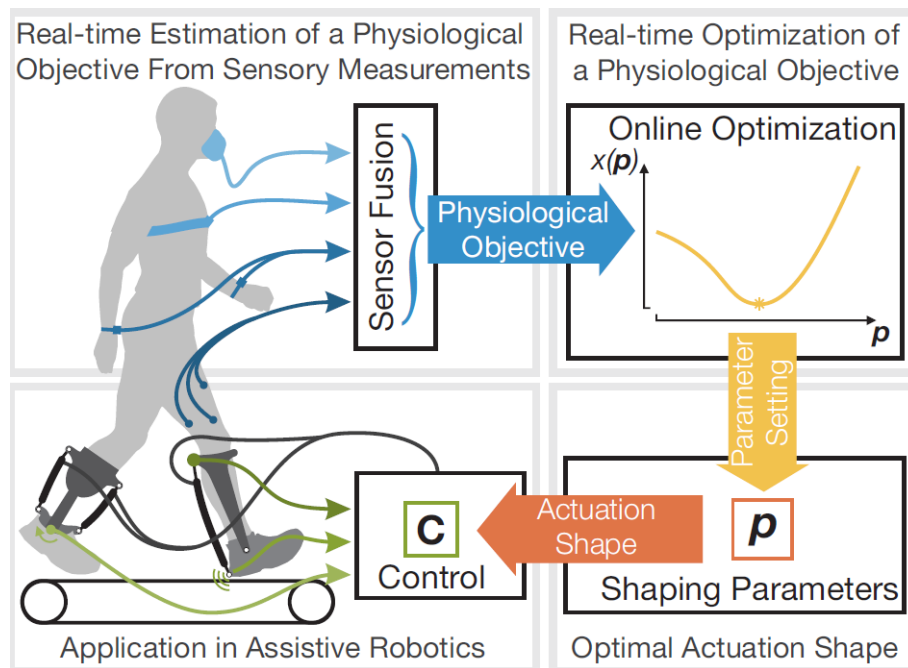


Figure 3: General control strategy of HITL algorithms. Taken from: (13).

This section will provide an overview of HITL algorithms currently being used on LLEs controllers. Additionally, it will be presented a discussion sub-section about the results found.

2.2.2 Methods

Search methodology

The literature search was conducted in the online database Scopus with the following combination of keywords: ("human-in-the-loop" OR "body-in-the-loop") AND (("lower limb exoskeleton" OR "ankle exoskeleton" OR "knee exoskeleton" OR "hip exoskeleton" OR "exosuit") OR ("exoskeleton" AND "energy cost" AND "metabolic cost") OR "exoskeleton" OR ("exoskeleton" AND "metabolic")). The search was executed between the 10th of October of 2022 and the 30th of August of 2023, and no restriction related to the papers' release date was taken into account.

Selection strategy

The papers were selected based on the following criteria: (i) presented a HITL algorithm implemented in an LLE; (ii) the HITL strategy fitted in the definition followed by this dissertation - online optimization of control parameters based on a physiological signal measured in real-time; (iii) presented an intelligible description of the control loop and optimization algorithm; (iv) performed human experiences and presented clear results. Additionally, a preference was given to papers: (i) published in

the last five years; (ii) with more citations; (iii) and published in journals with high impact factors.

Data extraction

The selected papers were analyzed in order to obtain the following information: (i) the type of exoskeleton controlled; (ii) the assisted motor task; (iii) the optimization algorithm and time required; (iv) the control parameters updated in real-time; (v) the physiological signal that controlled the optimization function (vi) the algorithms for MC estimation, sensors used for that matter, estimation time, and estimation error; (vii) the experimental protocol; (viii) the obtained results; (ix) and the limitations of HITL controllers.

2.2.3 Results

The search methodology resulted in 115 papers, that were filtered, following the mentioned selection strategy, and reduced to 17 articles. Tables 5 and 6 present the studies that developed a HITL strategy on an LLE based on the MC or based on another physiological signal, respectively. Tables 5 and 6 present the exoskeletons used by each study, the optimization algorithm and its optimization time, the control parameters, the MC estimation method and the sensors used for that purpose, the summarized experimental protocol, and the results obtained by each study.

The analyzed strategies differed significantly from each other, namely, regarding the optimization algorithm that was implemented and the cost function's signal. In the following text, these divergences will be presented.

HITL optimization algorithms

In 2016, Koller et al. (13) implemented a 1D gradient descent method that was able to find an optimal threshold for controlling the torque that was applied to an ankle exoskeleton. This study confirmed the feasibility of HITL strategies.

Only one year later, Zhang et al. (11) proposed another method for HITL optimization of the assistance of a tethered ankle exoskeleton to minimize the MC during walking, with a 4D covariance matrix adaptation evolution strategy (CMA-ES) optimization algorithm. CMA-ES optimizers have also been implemented in the control of a tethered ankle exoskeleton optimized for running (49), walking at self-selected speeds (12) and at different inclines (50), a tethered hip-knee-ankle exoskeleton for walking with heavy loads (51), walking at different speeds (52), and walking at different inclines (53), a tethered hip exosuit for optimized flexion torque while walking (54), and a portable hip exoskeleton (55).

Table 5: Studies conducted for the HITL optimization of the assistance of an exoskeleton, based on the user's MC

Study	Robotic device	Optimization algorithm	Opt. time	Control parameters	MC estimation algorithm	Sensors for MC estimation	Experimental protocol	Results
Koller et al., 2016 (13)	Tethered ankle exoskeleton	1D gradient descent method	50 min.	Ankle torque (on/off)	Discrete linear first-order system	Respiratory device (CareFusion Oxycon Mobile)	9 participants. 3 days of testing: first 2 days to map subjects' MC landscapes, last day to validate the optimized assistance	Average reduction of $18.0 \pm 4.6\%$ in MC
	Tethered ankle exoskeleton	Covariance matrix adaptation evolutionary strategy (CMA-ES)	64 min.	Ankle torque (value of peak torque, timing of peak torque, and rise and fall times)	First-order dynamic model	Respiratory device	11 participants. 1 day of testing Various locomotion conditions were tested afterward	Reduction of MC by $24.2 \pm 7.4\%$ compared to no-torque, and 14% compared to walking without the exoskeleton
Witte et al., 2021 (49)	Tethered active ankle exoskeleton emulator	Covariance matrix adaptation evolutionary strategy (CMA-ES)	60 min.	Ankle torque (value of peak torque, timing of peak torque, and rise and fall times)	First-order dynamic model	Respiratory device (COSMED)	11 participants 3 days protocol: habituation, optimization, and validation with 6 min. trials of running with different assistance strategies, repeated in reverse order afterward.)	Powered assistance improved energy economy by $24.7 \pm 6.9\%$ compared with zero torque and $14.6 \pm 7.7\%$ compared with running in normal shoes.
	Tethered passive ankle exoskeleton emulator			Emulated spring torque (max torque; shape constant; engagement angle)				Spring-like assistance was ineffective, improving energy economy by only $2.1 \pm 2.4\%$ compared with zero torque and increasing MC by $11.1 \pm 2.8\%$ compared with control shoes
Kanharaju et al., 2022 (56)	Tethered unilateral ankle exoskeleton emulator	Bayesian optimization method	15,8 min.	Two state-dependent (ascending/descending) parameters that are multiplied by the ankle angle to obtain the desired ankle torque	Phase-plane-based metabolic estimator in combination with a double deep-Q network for early stopping implementation	Respiratory device (K5, COSMED)	10 participants 2-day protocol: acclimation on the first day, optimization and validation on the second day. Validation with 4-min. squatting sessions (30 squats), of optimal assistance, no-torque and no-device conditions.	Reduction in the MC by 19.9% and <i>rectus femoris</i> muscle activity by 28.7% , compared to squatting without the exoskeleton

Table 5: Studies conducted for the HITL optimization of the assistance of an exoskeleton, based on the user's MC (Continued)

Study	Robotic device	Optimization algorithm	Opt. time	Control parameters	MC estimation algorithm	Sensors for MC estimation	Experimental protocol	Results
Ding et al., 2018 (19)	Tethered soft exosuit, with hip extension assistance	Bayesian optimization method	21.4 min.	Hip torque (peak and offset timing)	First-order dynamic model	Respiratory device (COSMED)	8 participants 1-day protocol, with the following sequence: 5 min. standing, 5 min. walking without the suit, 40 min. of optimization, 5 min. validation with optimal assistance and 5 min. of walking without the suit	Reduction of the MC by $17.4 \pm 3.2\%$ compared with walking without the device.
Kim et al., 2019 (57)	Tethered soft hip and ankle exoskeleton	Bayesian optimization method	< 60 min.	Hip force profile (onset, peak, and offset timing, peak magnitude, rising and falling force nodes) Hip force profile (onset and completion timings), and ankle force profile (peak and offset timings)	First-order dynamic model with Kalman filter	Respiratory device	2 participants 1-day protocol following walking experiments with and without the exoskeleton and with and without optimized parameters	Reduction of the MC by 35% and 7% compared with walking without the device.
Bryan et al., 2021 (51)	Tethered hip-knee-ankle exoskeleton emulator	Covariance matrix adaptation evolutionary strategy (CMA-ES)	234 min.	Hip, knee, and ankle assistance profiles (total of 22 parameters)	First-order dynamic model	Respiratory system (COSMED)	3 participants Each setting was optimized during 3 sessions, on separate days Validation was performed after each optimization in the following sequence: standing for 6 min., walking for 6 min., walking in a zero-torque mode for 10 min., and walking for 20 min. with optimal assistance, and then each action was repeated in reverse order	Exoskeleton assistance reduced the MC of walking by 48% with no load, 41% with a light load, and 43% with a heavy load, compared to wearing the exoskeleton without assistance Compared to walking without the exoskeleton, assistance reduced MC by 33% and 35% at 1.25 and 1.5 m/s, respectively
Franks et al., 2022 (53)	Active pelvis orthosis	Bayesian optimization method	6 to 18 min.	Hip torque (maximum extension, positive inflection, maximum flexion, and negative inflection)	Simulated human-APO model (OpenSim)	Reflective markers + a 12-camera motion capture system and GRF sensors	7 participants. 2-day protocol: model calibration, optimization, and validation Validation was done by 6 min. walks in each assistance mode, with 2 min. warm-ups, and 5 min. rests after 12 min. of testing.	Exoskeleton assistance reduced the cost of transport relative to walking without the exoskeleton by 37%, 40%, 42%, and 46% for 0°, 5°, 10° and 15° inclinations, respectively The lowest absolute MC was obtained by the HITL-optimized control scheme.

Table 6: Studies conducted for the HITL optimization of the assistance of an exoskeleton, that used physiological signals other than the MC

Study	Robotic device	Optimization algorithm	Opt. time	Control parameters	Characteristic optimized	Experimental protocol	Results
Jackson and Collins, 2019 (59)	Tethered bilateral ankle exoskeletons emulator	Heuristic co-adaptive control	30 min.	Ankle torque	Muscle activity and ankle kinematics	10 participants. Participants performed 30-min. trials in the adaptive and static conditions, one 6-min. without the exoskeleton, and two 6-min. trials with the exoskeleton in zero-torque mode.	9±12% reduction in MC when compared to normal walking
Yan et al., 2019 (60)	Tethered ankle exoskeleton (left foot)	Particle Swarm optimization	N/A	Ankle torque (value of peak torque and rise, fall, and peak times)	Muscle activity	N/A	Reduction in soleus muscles activity
Tucker et al., 2020 (61)	Powered hip, knee, and ankle exoskeleton	Gaussian process preference model with Bayesian optimization	N/A	Gait parameters (step length, duration, width, and maximum height, and pelvis roll, and pitch)	Gait preference (numerical feedback)	6 participants. First hour session was used to test users gait preferences. After 30 testing trials the subjects did the validation trial where their optimal gait was predicted according to their preferences	Gait preference predictions' accuracy was, on average, 83.3%
Han et al., 2021 (50)	Tethered ankle exoskeleton	Covariance matrix adaptation evolutionary strategy (CMA-ES)	24 min.	Ankle torque (value of peak torque, timing of peak torque)	Muscle activity (cost function)	5 participants. One participant walked under nine gait conditions, in order to optimize the cost function. The other 4 participants carried out the HITL optimization and validation	Compared to zero torque condition, optimized CF values under four gait conditions were reduced by 31.6% (normal), 33.2% (uphill), 26.2% (loaded) and 10.71% (incline)
Song and Collins, 2021 (12)	Tethered ankle exoskeleton	Covariance matrix adaptation evolutionary strategy (CMA-ES)	144 min.	Ankle torque (value of peak torque, timing of peak torque, and rise and fall times)	Self-selected walking speed	10 participants. 3-day experimental protocol. Familiarization session on the first day. On each of the second and third days, a 2x72-minute HITL optimization session followed by a validation session were conducted. Validation was done by walking at comfortable speeds in five assistance conditions	With torque optimized for speed, participants walked 42% faster than in normal shoes, while MC decreased 2% on average
Slade et al., 2022 (62)	Untethered ankle exoskeleton	Covariance matrix adaptation evolutionary strategy (CMA-ES)	32 min.	Ankle torque (value of peak torque and rise time)	Ankle angle, ankle velocity, and torque parameters for each control law	10 participants. 2-day protocol (optimization and validation) On the first day, participants performed one hour of walking in short bouts, in a real-world environment. On the second day, participants performed outdoor and treadmill validation tests - with each real-world and treadmill condition requiring about 15 and 8 min., respectively	Assistance optimized during one hour of naturalistic walking in a public setting increased self-selected speed by 9±4% and reduced the energy used to travel a given distance by 17±5% compared with normal shoes. Real-world Optimized assistance reduced the energy cost of treadmill walking by 16% at 1.25 m/s, 23% at 1.5 m/s, and 18% when walking up a 10° incline compared with Normal Shoes
Xu et al., 2023 (55)	Portable hip exoskeleton	Bayesian optimisation (BO) and Covariance matrix adaptation strategy (CMA-ES)	60 min. (BO) 90 min. (CMA-ES)	Hip torque profile (value of peak torque and peak, rise and fall times)	Muscle activity	4 participants. Participants walked with and without the exoskeleton, and in two control settings: no-assistance and HITL assistance. Participants walked for 25 seconds after adapting to each torque profile Two optimization algorithms were compared.	Muscle activity reduction of 18.48±31.62% and 21.12±19.64% for the BO and CMA-ES algorithms, respectively, compared to the no-assistance mode. Compared to not using the exoskeleton, the muscle activity increased 17.5±51.55% and 13.01±33.34% for the BO and CMA-ES algorithms, respectively.

In 2017, Kim et al. (11) evaluated the use of a Bayesian optimization (BO) algorithm for efficiently identifying control parameters that minimized the MC, estimated using respiratory data. Ding et al. (19) managed to use the BO method to optimize two hip torque parameters of a tethered soft exosuit, with a HITL control strategy, that allowed 17.4% of MC reduction (compared to not wearing the suit), taking only 21.4 minutes of optimization time. Other BO approaches have been developed, namely in the optimization of an active pelvis orthosis during normal walking (58), of a tethered unilateral ankle exoskeleton for squatting assistance (56), a tethered soft hip and ankle exoskeleton (57), and a portable hip exoskeleton during walking (55). More recently, in 2020, Tucker et al. (61) coupled a BO algorithm with a Gaussian process preference model to optimize the assistance of a powered LLE.

In 2019, Jackson and Collins (59) took a different approach to the HITL optimization problem and developed a heuristic co-adaptive controller, that used muscle activity measures of the *soleus* and *tibialis anterior* muscles and the ankle joint angle to control the magnitude of the ankle torque applied to a tethered bilateral ankle exoskeleton (59). Yan et al. (60), in 2019, used a Particle Swarm Optimization algorithm to optimize the assistance of another tethered ankle exoskeleton, with the objective of minimizing the muscle activity of the *soleus* muscles.

A recent study from Xu et al. (55) compared the results achieved by both CMA-ES and BO algorithms, with similar characteristics, in the optimization of four hip torque profile parameters to minimize the muscle activity of four participants. In the end, the optimization algorithms achieved similar optimal torque profiles but the CMA-ES resulted in a bigger reduction in muscle activity on average - a difference of almost 3%. However, the optimization times of the algorithms were not the same, as the CMA-ES lasted for one and a half hours (80 iterations) and the BO for an hour (61 iterations), taking 1.12 min./iteration and 0.98 min./iteration, respectively.

HITL cost function

The cost function implemented in the HITL algorithm is employed to allow the minimization of a kinematic or physiological signal related to the human perceived strain. The cost function's signal differed from study to study, but two main types of HITL algorithms could be distinguished: (i) studies that used the MC; and (ii) studies that used other signals, e.g. the users' muscle activity.

The majority of the analyzed studies used the indirect calorimetry method for the estimation of the MC, followed by: (i) a first-order dynamic model (8, 19, 49, 51–53, 57); (ii) a discrete linear system (13); or (iii) a phase-plane-based metabolic estimator, combined with a double deep-Q network for early stopping (56). Additionally, Kim et al. added an unscented Kalman filter to the first-order filter in order to obtain an optimal stopping point and, therefore, decrease the optimization duration. Alternately, Gordon et al.

(58) presented another methodology for estimating the MC of a user wearing an active pelvis orthosis by implementing a simulated human-device biomechanical model, in OpenSim, that was used to compute the MC based on the user trajectories and external forces. Table 5 presents the studies that developed HITL controllers with cost functions that minimized the estimated MC.

Alternatively, in 2019, Jackson and Collins (59) proposed a HITL ankle torque assistance based on the optimization of muscle activity and the ankle angle. In 2021, Han et al. (50) created a method based on a cost function of muscle activity, with weights optimized with a Swarm optimization algorithm, and Song et al. (12) used HITL optimization to control a tethered ankle exoskeleton and maximize the self-selected speed of users walking on a self-paced treadmill. Other approaches that did not optimize the MC of the participants were the studies of Yan et al. (60), which minimized the muscle activity of the *soleus* muscle, Tucker et al. (61), which optimized the gait according to the users' preference measured by a subjective feedback system, and Xu et al. (55) which minimized the muscle activity of the *rectus femoris* muscles.

A data-driven model was used by Slade et al. (62) to optimize the assistance of an untethered ankle exoskeleton in an outdoor environment, being the first HITL-controlled device used outside a laboratory. This team combined a data-driven model, that ranked different control laws after receiving the ankle angle, velocity, and torque parameters of each law, with a CMA-ES algorithm, that found optimal parameters and generated a new set of control laws to be tested. Despite not measuring the MC during the optimization in the real world, the data-driven was first trained with MC data obtained by indirect calorimetry, in a laboratory, using a respiratory device (Quark CPET, COSMED).

2.2.4 Discussion

This section presents a synthesis of the main conclusions taken from the literature review, as well as a presentation of the limitations of the current HITL controller. The first optimization algorithm used for HITL applications was the 1D gradient descent method. Because of its inefficiency and high sensitivity (13), other methods appeared. The CMA-ES algorithm allows for high-dimensional optimization problems, essential for the real-time control of complex exoskeleton joints (11). Despite the broad use of CMA-ES as the optimization algorithm in HITL strategies, this method is often time-consuming, due to the considerable number of iterations executed for parameter evaluations (11). Compared to the other strategies, the BO method allows for quicker optimization, even when metabolic measurements present high noise. However, this method assumes that the relationship between the assistance and the MC is not time-varying and is significantly more complex than the previous methods (11). Heuristic algorithms are simpler approaches, but the use of electromyography (EMG) sensors proved to be cumbersome to the user and limited the

use of this method to hip joint assistance (59). Additionally, the heuristic control strategy forces the users to walk with sub-optimal torque profiles for long periods, requires a rigorous characterization of the control parameters, and is probably ineffective for users with muscle impairments (59). A Particle Swarm Optimization algorithm has also been used for HITL assistance (60), however, the choice of this algorithm was not justified, and, at first sight, had no advantage over the other algorithms.

In terms of the signal being optimized, the vast majority of the papers used the users' MC, estimated by indirect calorimetry, as the cost function. However, these approaches are limited by the lengthy evaluation periods that are necessary to obtain the MC estimation, the low signal-to-noise ratio of the respiratory data, the difficulty of using respiratory masks in real-world applications, and the long delay between the instantaneous energetic demand and the physical respiratory measurements due to slow mitochondria response (8, 19, 56). Alternately, the use of biomechanical simulations to estimate the MC allowed for significantly quicker optimization times, but a comparison with the ground-truth MC, measured with respiratory data, showed significant differences between the two estimations (58). Besides, biomechanical simulations like the one presented at (58) often require precise joint kinematics and musculoskeletal knowledge about each specific user and significant computational power (14). Additionally, there were seven studies that did not use the MC cost as the cost function, using instead one alternative signal for the optimizations such as the muscle activity (EMG) (50, 55, 59, 60). These approaches, however, generally resulted in less MC reduction and focused on other results (12, 62).

Regarding the control parameters optimized in real-time, the papers generally focused on the optimization of the torque profile being applied to the controlled joint(s), with the exception of two studies that optimized: (i) two stage-dependant parameters (56) and (ii) six gait parameters (61). The studies that focused on the torque profile tried to simplify its curve by optimizing the minimum necessary parameters, namely the peak torque, peak timing, rise and fall times, and inflection points. Several of these papers noticed that the optimal values of some of the parameters, like the peak and fall times, varied little across participants, showing that some control parameters can actually be fixed without altering significantly the algorithm's performance (8, 12, 14, 51). One trend that could also be noticed in the analyzed studies was the implementation of HITL control on single-joint exoskeletons (mostly the ankle joint). Only three studies, performed by the same team, optimized the assistance of three joints (hip, knee, and ankle) at the same time (51–53).

As to the optimization time of the algorithms, it was possible to conclude that, as expected, increasing the number of control parameters also leads to an increment in the optimization period. The optimization time was roughly 10 to 15 minutes/parameter, which emphasizes the importance of the simplification

of the control parameters. However, this is not a linear relationship since the optimization time also depends on the type of parameters being optimized and the way the algorithm is initialized. Regarding the experimental protocol, most of the studies included 9 to 11 healthy participants and were performed in two to three days.

In regard to the results achieved by the analyzed studies, it was observed that all of the controllers were able to achieve a significant reduction of the minimized physiological signal, demonstrating the viability of HITL strategies. With respect to the studies that optimized the users' MC, reductions from 7% (57) up to 48% (53) of this signal were obtained, when comparing the optimized assistance to not wearing the exoskeleton. Regarding the controllers that did not optimize the MC, Xu et al. (55) achieved reductions of the users' muscle activity of 21%, while Slade et al. (62) obtained MC reductions of 23%, both when comparing the optimized controller to walking without the device. Additionally, better results were obtained when comparing the optimized assistance to a zero-torque condition, a strategy that increases human effort.

Limitations of current HITL control strategies

From the performed survey, several limitations and challenges associated with different HITL algorithms were observed, as follows. The optimization of multiple parameters per joint poses a significant limitation to the application of HITL strategies in fully actuated LLEs, as higher-dimensional problems require longer optimization times and greater computational power (8). Besides that, most of the used devices were tethered to the electric motors that supplied the power to their actuators, which shows the difficulty of implementing HITL controllers in portable systems (8, 13, 19). Additionally, HITL control strategies require a well-thought-out choice of the control parameters and their ranged values (12). Some studies also reported that the optimization is dependent on users' level of experience in using the devices and that adaptation effects are possible (19, 52, 53, 56).

In general, it was possible to conclude that optimization algorithms are already broadly used in HITL controllers and applied in LLEs in real-time, however, the analyzed approaches still heavily rely on the MC estimated by indirect calorimetry, which has several limitations and were only applied in tethered exoskeletons not designed for independent use in the real world. One approach to the MC estimation, without using respiratory sensors, is the implementation of regression models that use data obtained from portable and wearable sensors to establish a relationship between that data and the energy expended by the user. Despite not yet being implemented for real-time HITL optimization, these models will be presented in Chapter 2.3.

2.3 State of the art on metabolic cost estimation models

2.3.1 Introduction

To implement a HITL control strategy in exoskeletons it is essential to obtain the MC of the user in real-time. The gold standard method used to assess the MC during any activity is direct calorimetry, however, it is also the most expensive strategy, and it is impossible to implement on an exoskeleton being worn in the real world.

Two well-established estimation methods are used in the literature, as they obtain the closest results to the ground truth: indirect calorimetry and doubly labeled water. The former uses a respirometer device to measure the consumption of oxygen and the production of carbon dioxide, which can be used to obtain an approximated value of the MC. The latter uses labeled water, ingested by the human, that is traced until leaving the organism and used to evaluate how much carbon dioxide was used by the metabolism and estimate the MC of the participant (63).

Despite obtaining astonishing results, these methods require expensive equipment and materials, take too long to obtain the MC estimation, and most importantly are infeasible for everyday use by exoskeleton users. Therefore, a need for fast estimation methods based on data acquired in real-time by wearable, practical, and light sensors has arisen. Ideally, for HITL-controlled exoskeletons used in the real world (both for industry and/or daily living activities) these sensors should even be integrated into the device or built into a wearable electronic gadget, that is practical to carry around.

Several regression models have been created to estimate the MC, by establishing a relationship between sensor data and MC. Alvarez-Garcia (63) organized the possible strategies for MC estimation into three groups: single methods, activity-specific methods, and context-specific methods. The general proceeding followed by the estimation methods comprises the data acquisition and processing, features extraction and selection, and the computation of the MC estimate by the regression model(s) (63). The objective of this section is to provide an overview of current methods for the estimation of MC.

2.3.2 Methods

Search methodology

The literature search was conducted in the online database Scopus with the following combination of keywords: (("metabolic cost" OR "metabolic energy cost") AND "regression model") OR ("energy expenditure estimation") OR ("energy expenditure measurement" AND "machine learning"). The search

was executed between the 19th of October 2022 and the 10th of August 2023, and no restriction related to the papers' release date was taken into account.

Selection strategy

The papers were selected based on the following criteria: (i) presented a regression model capable of estimating the energy expended by a person based on wearable sensors; (ii) presented a clear description of the regression model; (iii) presented a concise description of the data used as input and the sensors used for its acquisition; (iv) presented at least one performance metric; (v) the experimental protocol studied at least one gait condition. Additionally, a preference was given to papers: (i) recently published; (ii) with more citations; (iii) and published in journals with bigger impact factors.

Data extraction

The selected papers were analyzed in order to obtain the following information: (i) the model's algorithm; (ii) the activities performed during the data acquisition; (iii) the model's input(s); (iv) the type and location of the sensors used for the data acquisition; (v) the estimation error and other performance metrics.

2.3.3 Results

The search methodology resulted in 202 papers, that were filtered, following the mentioned selection strategy, and reduced to 15 articles. Distinct models have been found in the literature, differing a lot from each other, as shown in Table 7. The studies varied in the activities performed, the algorithms implemented, the type of data collected for regression, and the number, type, and location of the sensors used. Early approaches focused on simpler models, like linear one-variable or multivariate regression models, based on one single data type. The algorithms constructed by Silder et al. (64) and Strath et al. (65) were based only on EMG signals and acceleration data, respectively.

Although simpler approaches, single-parameter methods present lower estimation accuracy, as one single parameter is unable to explain changes in the MC of participants performing a variety of tasks. Ingraham et al. (66) showed that even a simple multivariate linear regressor (LR) can achieve great results when a significant number of signals, obtained by six different sensors, are sent to the model. The team concluded that the more significant variables were the minute ventilation, EMG sum, waist acceleration, electrodermal activity, breath frequency, and HR; the signal that was least correlated with ground truth MC was skin temperature.

Table 7: Studies that proposed models for MC estimation using data acquired by wearable sensors. Legend: ACC - "Accuracy"; ASM - "Activity-specific method"; BMI - "Body mass index"; CSM - "Context-specific method"; MAE - "Mean absolute error"; N/A - "Not available/applicable"; SM - "Single Method"; SVM - "Support vector machine"

Study	Motor task(s) analyzed	Regression model	Type	Estimator signals	Other inputs	Sensors used	RMSE avg. (RMSE range)	Other metrics
Slider et al., 2012 (64)	Walking at different inclines	Multivariate regression model	SM	EMG	Treadmill inclination	EMG sensors	N/A	$R^2 = 0.96$
Pande et al., 2014 (67)	Standing, sitting, walking, ascending and descending stairs	Bagged regression trees	SM	Acceleration (waist) and barometer data	Weight, height, BMI, age, and gender	Built-in sensors of a smartphone	0.73 kcal/min (0.5-1.2)	$R^2 = 0.96$
Strath et al., 2015 (65)	Walking and daily living activities	Ngram time series	ASM	Acceleration (non-dominant wrist, hip, and ankle)	Weight, height, gender, fat %, and gait speed	Accelerometers	(0.84-1.04) MET	ACC = 89.24%
Sazonov et al., 2015 (68)	Standing, sitting, walking, running, and cycling	LR and SVM	ASM	Foot pressure and acceleration	Weight, BMI, log(BMI), and age	Five GRF sensors and one accelerometer on each foot	0.78 kcal/min (0.58-1.49)	N/A
Gjoreski et al., 2015 (17)	Walking, running, cycling, and daily living activities	Multiple contexts ensemble	CSM	Acceleration counts, HR, breath rate, chest, arm, and ambient temperatures, galvanic skin response, and activity	N/A	Accelerometers, zephyr sensor, and BodyMedia sensor	0.825 MET (0.57-1.2)	MAE = 0.601 MET
Zhu et al., 2015 (18)	Standing, sitting, walking, ascending and descending stairs	Convolutional neural network and a backpropagation multilayer perceptron	ASM	Acceleration (waist) and HR	Weight, height, basal metabolic rate, age, and activity level	Accelerometers and HR sensor	1.12 kcal/min (0.64-1.83)	N/A

Table 7: Studies that proposed models for MC estimation using data acquired by wearable sensors (Continued). Legend: ACC - "Accuracy"; ASM - "Activity-specific method"; IMU - "Inertial measurement unit; MAE - "Mean absolute error"; N/A - "Not available/applicable"; SM - "Single Method"

Study	Motor task(s) analyzed	Regression model	Type	Estimator signals	Other inputs	Sensors used	RMSE avg. (RMSE range)	Other metrics
Catal et al., 2018 (69)	Walking, running, cycling, and daily living activities	Boosted decision tree regression	ASM	Acceleration counts, HR, breath rate, chest, arm, and ambient temperatures, galvanic skin response, and activity	N/A	Accelerometers, zephyr sensor, and BodyMedia sensor	0.757 MET	MAE = 0.526 MET
Ingraham et al., 2019 (66)	Walking, running, cycling, and ascending stairs	LR	SM	Breath frequency and volume, acceleration (chest, hip, ankle), EMG, HR, electrodermal activity, skin temperature, and oxygen saturation	N/A	HR sensor, IMUs, wristbands, EMG sensors, and pulse oximeter	1.03 W/kg	N/A
Aziz et al., 2020 (70)	Standing, sitting, and walking	Support vector machine	ASM	Acceleration (wrist) gyroscope data (wrist), and activity	Walking speed	Smartwatch (right wrist)	N/A	MAPE = 15.37%
Sevil et al., 2020 (71)	Laying, sitting, running, cycling, resistance training, and daily living activities	Long short-term memory model	ASM	Acceleration (wrist), blood volume pulse, HR, skin temperature, galvanic skin response, and activity	N/A	Wristband	0.5 MET	ACC = 94.8%

Table 7: Studies that proposed models for MC estimation using data acquired by wearable sensors (Continued). Legend: ASM - "Activity-specific method"; ECG - "Electrocardiogram"; IMU - "Inertial measurement unit; MAE - "Mean absolute error"; N/A - "Not available/applicable"; SM - "Single Method"

Study	Motor task(s) analyzed	Regression model	Type	Estimator signals	Other inputs	Sensors used	RMSE avg. (RMSE range)	Other metrics
Slade et al., 2021 (15)	Walking, running, cycling, ascending, and descending stairs	LR	SM	Inertial measurements (shank and thighs)	Weight, height, and stride duration	IMUs	N/A	MAPE = 13.7%
Lucena et al., 2021 (72)	Laying, standing, sitting, walking, ascending, and descending stairs	Hierarchical LR	ASM	Acceleration counts (dominant wrist and ankle, and right hip) and HR	Weight, sex, and activity level	Accelerometers and HR sensor	0.613 kcal/min	R ² = 0.832
Lopes et al., 2022 (16)	Walking at different speeds, with and without exoskeleton assistance	Convolutional neural network	SM	Inertial measurements (feet, legs, pelvis, and torso), EMG, and HR	Weight, height, age, and gait speed	IMUs, HR monitor, and EMG sensors	0.36 W/kg	R ² = 0.79
Ni et al., 2022 (73)	Running at different speeds	Convolutional neural network and a two-stage regression module	ASM	Inertial measurements (waist) and ECG	Weight, height, waistline, sex, and age	IMU on the waist and 12-lead ECG sensors	0.71 kcal/min	R ² = 0.97 MAE = 0.53 kcal/min
Ramadurai et al., 2023 (74)	Squatting	Random forest	ASM	Magnitude of feet center of pressure and feet position	N/A	Pressure-sensing insoles	N/A	R ² = 0.79 MAE = 0.55 W/kg

In 2021, Lucena et al. (72), proposed a hierarchical regression method, a type of multivariate regression model, that started with fifteen different variables and was able to reduce them to only seven significant signals. The most significant variables were: the HR, wrist acceleration counts per minute, ankle acceleration counts per minute, participants' weight, physical activity level, and participants' gender.

More complex approaches have been implemented with machine learning algorithms such as bagged regression trees and support vector machine (SVM) models, like the ones implemented by Pande et al. (67) and Sazonov et al. (68), respectively. In 2015, Gjoreski et al. (17) went a step further and compared a novel multiple contexts approach with five other models: (i) a multilayer perceptron feedforward artificial neural network (NN), (ii) an SVM, (iii) a multiple LR, (iv) a gaussian process regressor (GPR), and (v) a model tree. Additionally, the authors compared two different aggregation techniques: average and median. The multiple contexts algorithm consisted of a congregation of eight regression models, each trained for a different feature, and obtained better results than any other method when the median aggregation approach was implemented.

Three years later, Catal et al. (69) used the dataset published by Gjoreski (17) and studied the performance of several regression-based machine learning models: a Bayesian LR, a boosted decision tree (BDTR), a decision forest, an LR, an NN, and a Poisson regressor. The BDTR model outperformed all algorithms, including Gjoreski's multiple context algorithm.

Deep learning algorithms have recently also been used for MC estimation. This approach, despite being more complex, ends up reducing the computational load by enabling automatic feature selection. Zhu et al. (18) used a convolutional neural network (CNN) and, despite using just two simple sensors, it was able to outperform an activity-specific LR and a backpropagation multilayer perceptron artificial NN. In 2020, Sevil et al. (71) used a long short-term memory model to estimate the MC, which achieved a better performance than six other machine learning models. More recently, in 2022, Ni et al. (73) implemented a CNN, followed by a two-stage regression layer, and were able to obtain even better results by using electrocardiogram (ECG) sensors. Additionally, Lopes et al. (16) concluded that CNNs also outperform long short-term memory networks.

Regarding the type of signals more commonly used by regressors, acceleration measurements stand out, since thirteen of the fifteen papers analyzed used at least one acceleration or inertial sensor (15–18, 65–73). However, the articles differentiated themselves by the number of acceleration sensors used and their location. Six of the papers only used one sensor, five used two or three sensors, and three studies used four to five different sensor locations. The locations more common for the acceleration measurement

were the wrist and ankle (six papers), followed by the waist, thigh, and chest (three papers). Additionally, the hip, foot, shank, leg, pelvis, and torso were also used for the location of the accelerometers, but less frequently. Besides the acceleration, the HR (16–18, 69, 71, 72) and activity levels (17, 18, 69–72) were also commonly measured, followed by the EMG (16, 64, 66), the breath rate (17, 18, 66), skin temperature (17, 18, 71), galvanic skin response (GSR) (17, 18, 71), ambient temperature (17, 18), breath volume (66), electrodermal activity (66), oxygen saturation (66), ECG (73), foot center of pressure (74), foot position (74), blood volume pulse (71), and gyroscope data (70).

Using more signals to estimate the MC can be useful to improve the regression models' performance, however, it generally results in a need for a larger network of sensors. Some studies have successfully managed to obtain various signals with only one sensor. Pande et al. (67) used only one smartphone to measure the waist acceleration and air pressure. Aziz et al. (70) used a smartwatch, placed on the right wrist, and measured the 3D acceleration and the 3D rotational data. Ramadurai et al. (74) used a pressure-sensing insole to measure the foot center of pressure and its position, and managed to obtain various features from that data, like the mean magnitude and its standard deviation, the minimum and maximum pressure, and others.

2.3.4 Discussion

Several different regression models were implemented across the analyzed papers, however, none of the studies were performed exactly with the same conditions. From the papers that studied multiple regression models, it was possible to conclude that the BDTR from (69) outperformed a multiple context algorithm, a multilayer perceptron feedforward artificial NN, an SVM, a multiple LR, a GPR model, a model tree, a Bayesian LR, a decision forest regression, an LR, a NN regression, and a Poisson regression (17, 69). Additionally, a long short-term memory network achieved better results than an ensemble learning model of decision trees, a k-nearest neighbors model, a linear discrimination model, an SVM, and a decision tree model (71). Finally, CNNs were able to outperform a long short-term memory network (16), an activity-specific LR (73), and a backpropagation multilayer perceptron artificial NN (73).

Studies that compared different input signals in order to analyze which ones were the significant variables with a higher correlation to the ground truth were also scarce in the literature. From the two studies that performed this analysis - (66) and (72) - it was possible to see that the acceleration signals measured at the waist, wrist, and ankle were more significant than the acceleration of the chest and hip (66, 72). Additionally, the minute ventilation data, EMG composite sum, electrodermal activity, breath frequency, and HR were more significant than the breath volume, skin temperature, and oxygen

saturation (66), and the HR and physical activity level were more significant than the anxiety level (72). Regarding the anthropometric features given as an input to the regression models, the subjects' body mass, and gender were also deemed as more significant than the ethnicity, height, age, and body composition (72).

As seen in Table 7, all of the analyzed studies produced low RMSEs on average (all were below 1.12 kcal/min (18)), however, the errors obtained varied significantly depending on the activity level, as seen by the range of values presented (17, 18, 65, 67, 68). The results obtained from the studies are difficult to compare since they differed a lot from each other regarding the tasks analyzed, the algorithm, the number of regressor signals, and the sensors type and location, but, in general, the error range increased when activities with greater MC consumption, such as running and cycling, were introduced to the participants (17, 68).

In general, it was possible to conclude that the results from models that estimate the MC can be improved by using algorithms like BDTRs or CNNs and using the most significant variables (like the waist, wrist, and ankle acceleration, HR, physical activity level, and EMG). Despite the significant advances made in MC estimation algorithms, none of the studies discussed so far were orientated specifically for HITL optimization, and most of them were even unfeasible for that application because of the impractical nature and bulkiness of the sensors used. To the authors' best knowledge, only one study performed MC estimation of participants wearing an exoskeleton (Lopes et al. (16)). This article represents the starting point of this dissertation.

2.4 General conclusions

This sub-chapter presents a synopsis of the conclusions that were taken by the literature review, namely the limitations of the current industrial LLEs and HITL controllers.

Firstly, the state of the art demonstrated that several conditions still hinder the application of LLEs in industrial contexts, despite their potential for minimizing workers' physical stress and strain. Those limitations include the lack of human intention recognition algorithms, the lack of studies that analyze the efficacy of wearing the exoskeletons on industrial sites, and the lack of exoskeletons using adaptive controllers to tailor, in real-time, the exoskeleton assistance according to worker's real physical needs and effort.

Despite the advantages of adaptive controllers like HITL strategies, these also implicate some major problems. This dissertation aims to address the lack of applications of HITL controllers for exoskeleton-driven assisted work.

The major limitation of HITL controllers that use the MC as the optimization function is that this physiological signal needs to be estimated in real-time, and as fast as possible. The standard method for this procedure is indirect calorimetry, however, this requires expensive and not practical sensors (respiratory masks), generates signals with a low signal-to-noise ratio, and takes too long to obtain the MC estimation. One alternative for calorimetry estimation is the use of regression models, such as LRs and CNNs. However, these methods have still not yet been incorporated into HITL controllers. This dissertation aims to tackle this challenge. Another major limitation is the required time for the optimization of the exoskeleton assistance, which hinders the use of this strategy in real-world applications. To overcome this issue, this dissertation aims to only optimize the parameters that mostly vary across subjects to limit the optimization time. Additionally, no HITL controller has been developed solely for knee assistance, and all of the controlled devices were tethered to the actuator. This dissertation aims to implement the HITL controller for an untethered knee exoskeleton.

3 System requirements and overview for assisted working

This chapter begins by introducing the requirements for industrial LLE devices that assist workers during the relevant occupational tasks, as previously discussed: (i) carrying loads; (ii) lifting loads; and (iii) stationary assembly tasks. The analysis of these tasks regarding the workers' joints' ROM and torque ranges is presented. Then, the lower-limb powered exoskeleton used in this work - SmartOs - is presented. This chapter follows with the presentation of the proposed controller for this dissertation, which aims to tackle the major challenges identified in Chapter 2.

3.1 Industrial assistance requirements

As mentioned in Chapter 1, the industrial tasks that are more often associated with WMSDs on the lower limbs are carrying and lifting heavy loads (Figures 4a and 4b) and tasks that require stationary positions for long periods of time (Figures 4c and 4d). Table 8 associates a motor task (human motion/position) to each of these industrial tasks and presents examples of industrial sectors where each task is more prevalent.



(a) Carrying loads. (75)

(b) Lifting loads. (76)

(c) Sitting. (77)

(d) Standing. (78)

Figure 4: Industrial tasks more often associated with WMSDs.

These industrial tasks will be the focus of this work which was developed using an LLE designated by SmartOs. As a preliminary step, it was conducted a survey of the humans' needs during the corresponding motor tasks, both in terms of the natural joints' ROM and required joint torques. Table 9 presents the average ROM for each lower limb joint during unloaded walking (79), loaded walking (79), squatting (80), sit-to-stand (81), and stand-to-sit (81) motions and the SmartOs' features.

Table 8: Industrial tasks more often associated with WMSDs and their corresponding motor tasks, and the industrial sectors with a significant prevalence of those tasks.

Industrial task	Human motor task	Examples of industrial sectors
Stationary assembly tasks	Standing and sitting	Assembly lines, production lines, and textile
Load carrying	Walking	Logistics, construction, mining, transportation
Load lifting	Squatting	

Table 9: Natural ROM of each joint during each of the relevant motor tasks, and the correspondent exoskeleton limits (79–81)

Motor task	Hip ROM (°)	Knee ROM (°)	Ankle ROM (°)
Walking (unloaded)	[-20,25]	[10,70]	[-15,15]
Walking (20kg loads)	[-10,30]	[10,70]	[-10,15]
Squatting	[-20,80]	[0,125]	[5,40]
Sit-to-stand/Stand-to-sit	[10,90]	[0,90]	[5,25]
Total	[-20,90]	[0,125]	[-15,40]
SmartOs ROM	N/A	[-3,100]	[-20,20]

Table 9 reveals that the SmartOs' ROM is high enough for the motor tasks analyzed, with the exception of the squatting motion, which is limited due to a constraint in the knee and ankle joints during the flexion movement. This restriction however is not expected to limit humans' dexterity and the users will still be able to lift loads.

Table 10 presents the average normalized torque range for each lower limb joint during unloaded walking (79), loaded walking (79), squatting (80), sit-to-stand (82), and stand-to-sit (82) motions, along with the total torque range required for a 100 kg person.

Table 10 shows that the LLE used in this work (SmartOs) provides a 100 kg person with all the necessary torque during all the discussed activities since it can achieve up to 180 Nm of peak torque. Additionally, Tables 9 and 10 established that the hip and knee joints are of critical importance, as they generally perform wider ROMs and require larger torques (especially in the extension motions). Having this in mind, going forward, this dissertation will focus on the assistance of the knee joint because of its crucial importance during all the mentioned industrial tasks.

Table 10: Natural torque of each joint during each of the relevant motor task (79, 80, 82)

Motor task	Hip Torque (Nm/kg)	Knee Torque (Nm/kg)	Ankle Torque (Nm/kg)
Walking (unloaded)	[-1,1.2]	[-0.2,0.8]	[-0.1,1.4]
Walking (20 kg loads)	[-0.8,1.8]	[-0.3,1.4]	[-0.1,2.0]
Squatting (15 kg loads)	[0,1.8]	[-1.5,-0.2]	[0.2,1]
Sit-to-stand/Stand-to-sit	[0,0.4]	[0,0.5]	[0.2,0]
Total	[-1,1.8]	[-1.5,1.4]	[-0.1,2.0]
Total (100 kg person)	[-100,180] (Nm)	[-150,140] (Nm)	[-10,200] (Nm)

3.2 SmartOs system

SmartOs is a wearable, modular, lower-limb powered exoskeleton developed to assist users according to their needs and intentions. The system is composed of two active joints on the right leg - an ankle and knee module - with one DOF per joint in the sagittal plane. The system follows a modular architecture, enabling the inclusion of further active joints. Each joint is coupled with a potentiometer, four strain gauges (that measure the human-robot interaction torque), and a Hall effect sensor (that measures the actuators' torque). The SmartOs device also accommodates two force-sensitive resistors in the heel and toe of each foot and three inertial measurement units (IMUs) (83).

The actuators consist of DC motors (EC60-100W, Maxon) coupled to gearboxes (CSD20-160-2A, Harmonic Drive), from the company 'Technaid', and provide average torques of 35 Nm and peak torques of 180 Nm. The system is powered by a lithium iron phosphate battery with an autonomy of 8 hours. The system weighs approximately 5.5 kg (considering both ankle and knee joints) and can assist users with heights ranging from 150 to 190 cm and weights ranging from 45 to 100 kg. Its assistance is limited to gait speeds from 0.5 to 1.6 km/h (83). Figure 5 presents the knee module of the SmartOs system and its sensors and components.

The SmartOs is controlled by a non-centralized team-developed architecture that is structured in three control levels (high-, mid-, and low-level). The frequency of the low-level and mid/high-level controllers is 1 kHz and 100 Hz, respectively (83). The architecture includes: (i) a central controller unit (CCU) responsible for running the system's high-level control, gait analysis algorithms, and external communications; (ii) a low-level orthotic system (LLOS) responsible for running the low- and mid-level control of the actuators; (iii) a wearable motion system (WMSS) responsible for the communication with external wearable and non-intrusive sensors systems; and (iv) a mobile application (APP) that works as the user-device interface

and allows the system configuration and initiation (83). Figure 6 presents a diagram of the architecture.

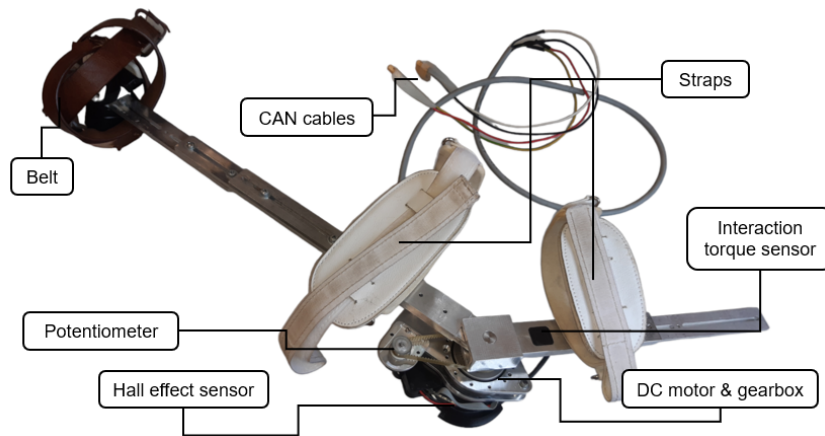


Figure 5: Knee module of SmartOs and respective components.

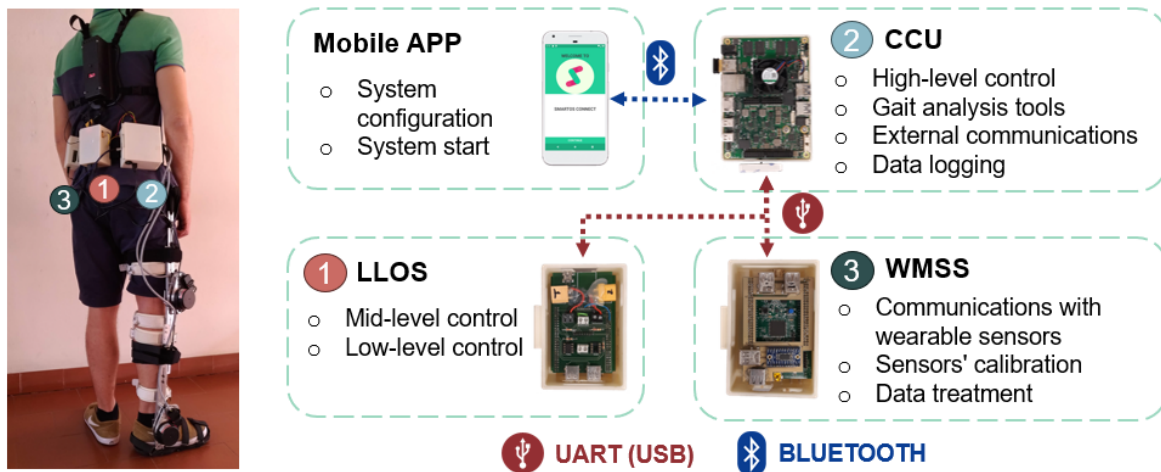


Figure 6: Diagram of SmartOs architecture.

The CCU is a UDDO X86 board, with a quad-core central processing unit (64 bits) and 8GB of RAM. This board interfaces with both LLOS and WMSS boards by a UART protocol using USB cables and with the mobile APP through Bluetooth. The high-level code is executed in the CCU, in an Ubuntu operating system, in C++. The code flow is organized using threads with different priority levels running in parallel to each other (83). The CCU code is developed using the QT Creator integrated development environment (IDE).

Both LLOS and WMSS boards are STM32F4-discovery boards (STMicroelectronics, Switzerland), with STM32F407VGT microcontrollers (32 bits). These boards are coupled with USB converters (FT232RL FTDI), allowing direct communication with the CCU by USB. The software of these boards is executed in C. These boards' code is developed using the freeRTOS operating system, which allows easy management

of the different threads and tasks, in the Keil uvision 5.0 IDE. The LLOS board is also able to send command controls to the active actuator through a CAN protocol, at a frequency of 1 kHz. (83).

The WMSS can connect several different sensor systems to the CCU. This work used a wearable inertial sensor system, composed of IMUs, called InertiaLab. Each IMU is comprised of an MPU-6050, a small and light module capable of measuring the 3D acceleration and the 3D rotational rate. The WMSS board communicates with these IMUs by an I²C (Inter-Integrated Circuit) protocol. Figure 7a presents the InertiaLab's architecture, and Figures 7b and 7c present a single IMU, displaying the mini USB port and the MPU-6050 module (the blue board) (84).

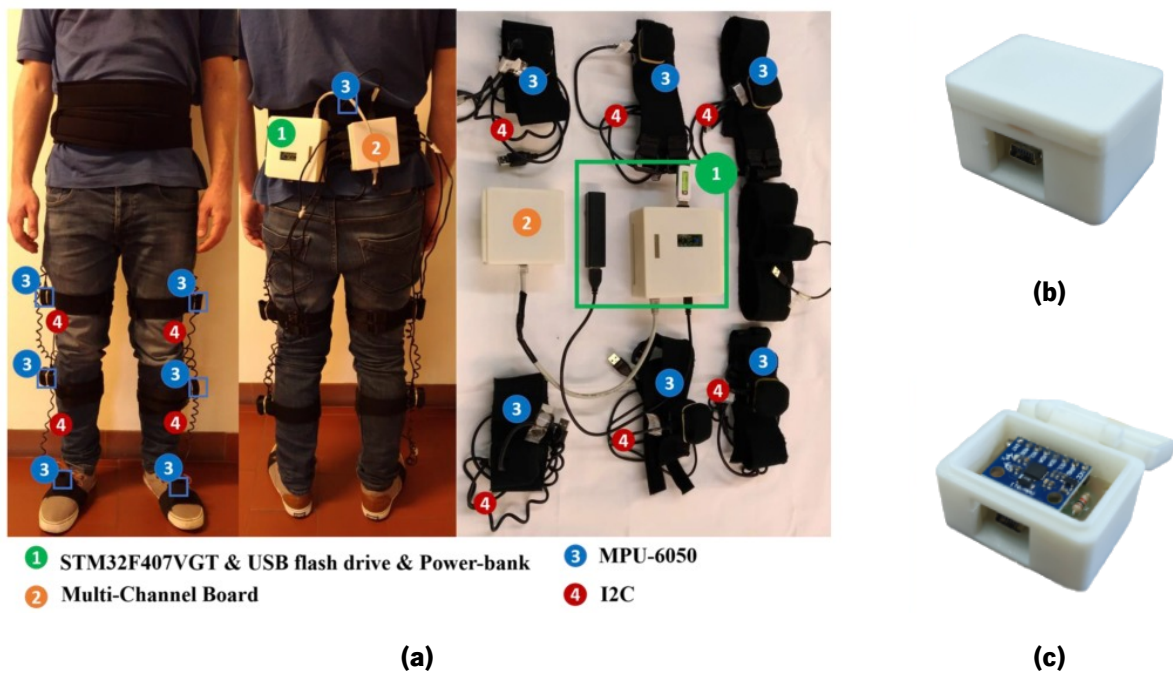


Figure 7: *InertiaLab system, where (a) presents the full InertiaLab's architecture (84), and (b) and (c) present a single IMU with the lid closed and opened, respectively.*

3.3 Proposed solution

This section presents a proposed solution to the problems identified in the state of the art, that was conceptualized envisioning the SmartOs' knee module. Chapter 2.1 presented the various limitations of current industrial exoskeletons. It was seen that adaptive controllers according to the human physical effort have not been implemented yet in industrial exoskeletons. This work aims to develop an adaptive control strategy capable of automatically optimizing the exoskeleton's control parameters in real-time by minimizing the user's effort - a HITL controller. The HITL controllers require real-time MC measurements, which, as seen in Chapter 2.2, is commonly estimated by indirect calorimetry. However,

the indirect calorimetry method faces major issues, namely with the materials cost, the impracticality of the respirometer device, the noisy nature of the estimation, and the time requirements for estimation. Chapter 2.3 presented various machine and deep learning models capable of estimating the MC based on wearable and non-intrusive sensors' data. However, most of these strategies are not feasible for HITL assistance of industrial workers.

The proposed solution consists of the implementation of a HITL controller on the knee actuator of SmartOs, that adapts the knee torque according to the users' MC and interaction torque. The MC is estimated by a regression model based on wearable and non-intrusive sensors' data. Figure 8 shows the general diagram of the proposed HITL control strategy for the SmartOs device, that aims to tackle these current challenges.

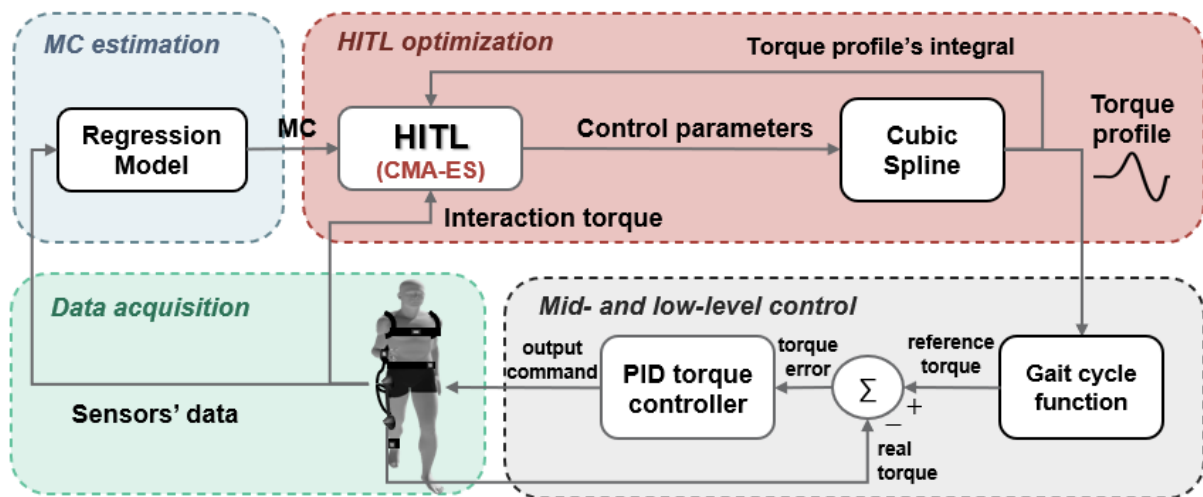


Figure 8: Diagram of the proposed controller for optimizing the assistance of an active knee exoskeleton and minimizing the MC of a person in real-time.

The HITL controller uses a regression model to estimate the MC (*MC estimation*, blue block of Figure 8) based on real-time wearable and non-intrusive sensors' data measured while the human is performing the motor task (*Data acquisition*, green block of Figure 8). The development and validation of this MC estimation model are presented in Chapter 4. The controller then tries to minimize the user's effort by adapting two control parameters: the magnitude of the peak flexion and extension torques of the knee joint. This optimization is performed by the evolutionary algorithm CMA-ES in real-time (*HITL optimization*, red block of Figure 8). This optimizer changes the peak torques, in real-time, and analyzes the effect of the new control parameters on the MC of the person. Additionally, the CMA-ES algorithm also analyses the reference torque's integral and the cumulative sum of the interaction torque (torque resultant from the interaction between the user and the device) at each iteration. The algorithm ends when the optimal torque

is found, i.e., the peak torque magnitudes that lead to a minimal MC, interaction torque, and reference torque's integral, or when any termination condition is verified.

Regarding the HITL controller, a natural cubic spline interpolator is used to generate a different torque profile for every set of control parameters from CMA-ES (*HITL optimization*, red block of Figure 8). This interpolator receives a set of torque values at various gait cycle percentages, two of which are variable: the peak flexion and the peak extension torque magnitudes. Each generated torque profile is then sent to a proportional-integral-derivative (PID) torque controller. The development and validation of the HITL controller are demonstrated in Chapter 6.

For implementing the HITL into the SmartOs' architecture, it was also necessary to develop and implement a novel PID torque controller (*Mid- and low-level control*, grey block of Figure 8). The controller receives a reference knee torque profile and drives the active actuator to follow the desired pattern. The mid-level stage of the controller estimates the gait cycle phase and obtains the reference torque for the respective phase. Then, a PID controller ensures that the system's knee joint torque reaches the desired torque, at the low-level block of the controller. The development and validation of this torque controller are demonstrated in Chapter 5.

4 Metabolic cost estimation

This chapter describes the development of the regression model used to estimate the MC in real-time based on the data from wearable and non-invasive sensors. The work presented in this section started with the analysis of publicly available datasets constituted by data from a variety of wearable sensors and the MC of the participants estimated through indirect calorimetry, by measuring the oxygen consumption and carbon dioxide production with a respirometer device. To choose the best dataset different requirements were established and the dataset used to train and benchmark the models for MC estimation was chosen based on those conditions.

One focus of this study was the preprocessing of the data. The various phases during the data preparation were studied and optimized to obtain the best possible model performance. This was followed by the selection of the best estimators, i.e., the input signals that had a higher correlation to the MC and could be used to estimate it accurately. Additionally, activity-specific models, i.e., models that were trained using data from only one motor task, were compared to general models that were trained with all the data from different activities.

Various machine and deep-learning models were explored and compared. The regression models included: (i) linear models; (ii) decision trees; (iii) SVMs; (iv) Gaussian support vector machines (GSVMs); (v) GPRs; (vi) kernel approximations; (vii) tree ensembles; (viii) simple NNs; and (ix) CNNs. The models were evaluated regarding the RMSE and the coefficient of determination (R^2) during the validation phase, which followed the leave-one-out cross-validation (LLOCV) method. The best model was then used to estimate the MC of a test participant whose data was not used during the training and validation processes.

Afterward, the best regression model was integrated into SmartOs' architecture to enable the MC estimation in real-time based on data acquired by the system. This implementation was then validated with a real-time protocol where the MC estimation made by the regression model was compared to the estimation made by a respirometer, through indirect calorimetry.

4.1 Introduction

Indirect calorimetry is the most common method used to estimate the MC of a person during a certain activity. This method computes an MC estimation by applying an equation that linearly relates the consumption of oxygen and carbon dioxide production to the user's MC. The gold standard equation used for this purpose is Brockway's equation, first published in 1987 (85).

Theoretically, indirect calorimetry is very simple to apply due to the simplicity of the equation and the

reduced number of variables it requires, however, in practice, measuring the oxygen and carbon dioxide flow in the human body is not elementary. The rate of production and consumption of these gases can be measured using a respirometer device, like the one presented in Figure 9 (63).



Figure 9: Woman wearing a respirometer device (K5, Cosmed, Italy). Taken from: (86)

Despite facilitating the MC estimation with high accuracy, these devices, as seen in Figure 9 are not practical to wear, as breathing through a mask can be uncomfortable for long periods of time, and the person is required to also wear a heavy system, on their back, that measures the gas exchanges. Additionally, these devices are also quite expensive and have a limited autonomy of only a few hours, depending on the device.

Furthermore, the limitations associated with the indirect calorimetry method are not just related to the respirometers. Another significant drawback of this method is the time it requires to estimate the first value of the MC for a certain activity - roughly 3 minutes (87). This can be explained by the fact that the human body takes some time to adapt to a change in activity, resulting in a time period during which the MC is not yet stable (i.e., the MC is not yet in a steady-state).

An alternative to indirect calorimetry is the use of regression models that estimate the MC of a person based on data acquired by wearable and non-intrusive sensors placed on the person. Regression models work in a similar way to indirect calorimetry, in the sense that they find an equation, or equation system, between the MC and a set of variables measured in real-time. Regression models replace the need for

respirometers since they can form a correlation between the MC and signals like acceleration, HR, and the GSR, which are much more practical to measure (17, 65, 67). Furthermore, this method can substantially decrease the time required to obtain an MC estimation.

Despite the expanding studies developing regression models for MC estimation (analyzed in Table 7), these models have not been developed for HITL control strategies in exoskeletons and much less for assisting industrial workers and measuring the MC while they perform their tasks. When developing a system meant to be worn by workers, it is important to use the smaller number of sensors as possible, that these sensors are light and small, do not affect the natural movement of the workers, and are easy to don and doff. Therefore, it is critical that the regression model achieves high accuracy based on a limited number of practical wearable and non-intrusive sensors. The regression model presented in this chapter aims to tackle these challenges.

4.2 Methods

4.2.1 Dataset selection

Several datasets for estimating the MC are available online, however, they all differ significantly from each other either in terms of the number of participants, the tasks performed during the experimental protocol, or the signals measured. Therefore, the datasets were compared with each other according to a set of desired requirements. The requirements for the dataset were: (i) participants must have performed, at least, the sitting and walking activities, as they are activities similar to the motor tasks identified in Chapter 3.1; (ii) the model's inputs contained acceleration and HR signals, as they can be measured by practical sensors for industrial applications. Additionally, a preference was given to datasets with the most activities analyzed and with the greater number of participants (and age variability). Table 11 presents the publicly available datasets and their characteristics.

From Table 11, it was possible to observe that Cvetković et al. (88) and Ingraham et al. (66) datasets (colored in orange) were the only ones whose participants performed both the walking and sitting activities. Another strong suit of these datasets is the fact that participants also performed the standing activity. Additionally, both the acceleration and HR were provided, and the acceleration was measured in four different on-body locations. However, a negative aspect of both datasets was the low number of participants in each study and the low diversity of the participants' ages.

Table 1.1: Publicly available datasets used for estimating the MC. Legend: EEG - "Electroencephalogram"; PPG - "Photoplethysmography"; ST - "Skin temperature"

Authors	Participants	Activities performed	Estimator signal(s)	Sensors used
Cvetković et al. (88)	10 (27.2±3.1 yr)	Daily living activities, walking, sitting, standing, cycling, and running	Acceleration (ankle, thigh, chest, wrist, thigh pocket), HR, Breath rate, ST and R-R interval, near-body temperature, ST (arm), GSR (arm)	Shimmer, Zephyr BioHarness, BodyMedia FIT, Cosmed K4b2, Smartphone
Gashi et al. (89)	17 (30±11 yr)	Sitting, standing, cycling and running	Acceleration (ear, head, chest, wrist), HR (wrist, chest), angular velocity (ear, head), PPG, electrodermal activity, skin temperature, ECG, respiration rate, blood volume pulse, HR variability, EEG, breath-to-breath interval	Nokia Earbuds, Empatica E4, Zephyr Bioharness, Wahoo Tickr, Apple Watch, Fitbit Sense, Muse S
Ingraham et al. (66)	10 (27.4±4.5 yr)	Standing, Sitting, Walking, running, cycling, and ascending stairs	Breath frequency and volume, acceleration (chest, hip, ankle, wrist), EMG, HR, electrodermal activity, skin temperature, and oxygen saturation	Respirometer, HR sensor, IMUs, wristbands, EMG sensors, and pulse oximeter
Slade et al. (2019) (14)	10 (33.7±9.0 yr)	Walking with ankle exoskeleton, walking with loads and in incline conditions	EMG (7 muscles per leg) and GRF	Wearable EMG electrodes (Delsys) and pressure sensing insoles.
Slade et al. (2021) (15)	13 (23.8±2.6 yr)	Walking, backward walking, hopping, loaded walking, and running	EMG (7 muscles, left leg), GRF, acceleration (shank and thigh) and HR	Electrodes, GRF sensors, HR monitor (chest), IMUs
	10 (24.5±2.5 yr)	Walking, backward walking, hopping, loaded walking, running, ascending stairs and cycling	Acceleration (shank and thigh)	IMUs
	24 (34.8±11.6 yr)	Walking, running, ascending stairs and cycling		

Therefore, both these datasets were first selected for a preliminary analysis. During this study, it was noticed a significant disparity between the 3D acceleration signals of the two datasets. A further investigation demonstrated that Cvetković's acceleration data did not match the expected values, therefore, that dataset was set aside and Ingraham's dataset was chosen to train, validate, and test the regression model.

Ingraham's dataset

Ingraham's dataset was obtained at the University of Michigan, in Ann Arbor, U.S.A. The data, obtained from ten participants, is publicly available in figshare (90). The participants (8 male and 2 female) had ages between 24 and 37 years old ($27,4 \pm 4,5$ yr), with body masses between 58.05 kg and 95.24 kg (69.1 ± 9.9 kg), and body heights between 1.63 m and 1.85 m (1.76 ± 0.09 m). Figure 10 presents the location and the signals measured by each sensor worn by the participants.

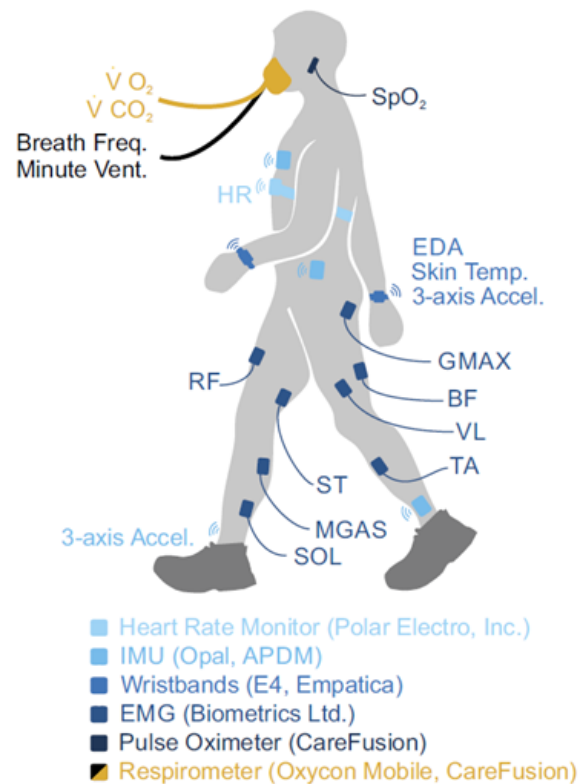


Figure 10: Representation of on-body location of the sensors of Ingraham's dataset. Legend: BF - "Biceps femoris"; EDA - "Electrodermal activity"; GMAX - "Gluteus maximus"; MGAS - "Medial gastrocnemius"; RF - "Rectus femoris"; SOL - "Soleus"; SpO₂ - Oxygen saturation; ST - "Semitendinosus"; TA - "Tibialis anterior"; $\dot{V}CO_2$ - Rate of carbon dioxide consumption; $\dot{V}O_2$ - Rate of oxygen consumption; VL - "Vastus lateralis" (66).

Each person wore 25 different sensors: (i) four 3-axis accelerometers on the chest, waist, and the right and left ankles (Opal, APDM, U.S.A.); (ii) two wristbands on the right and left wrists (E4, Empatica, Italy);

(iii) sixteen surface electrodes placed on eight different muscles on each leg; (iv) a pulse oximeter (Oxycon Mobile, Carefusion, U.S.A.); (v) an HR monitor (Polar Electro, Finland); and (vi) a portable respirometer (Oxycon Mobile, Carefusion, U.S.A.).

The participants performed six activities: (i) leveled walking at 2.16, 3.24, and 4.32 m/s; (ii) inclined walking at 2.16, 3.24, and 4.32 km/h; (iii) backward walking at 0.4 and 0.7 km/h; (iv) running at 6.48, 7.92, and 9.72 km/h; (v) cycling at 70 and 100 rpm, with 3 different levels of resistance; and (vi) stair climbing at 60, 75, and 90 Watts. Additionally, for each activity, the participants also stood and sat down for 6 minutes, before and after the activity, respectively.

4.2.2 Data preprocessing

The first step was the data preprocessing, a fundamental procedure in training machine and deep learning models. Firstly, the ground-truth MC of Inghram's dataset was computed by indirect calorimetry using Brockway's equation (Equation 1), where $\dot{V}O_2$ and $\dot{V}CO_2$ are the rates of consumption of oxygen and production of carbon dioxide, respectively.

$$MC (W) = 16.58 \dot{V}O_2 (mL/s) + 4.51 \dot{V}CO_2 (mL/s) \quad (1)$$

Then, it was analyzed if normalizing the HR and MC by the weight of the participants improved the MC estimation's accuracy (16, 17). This was done by dividing the two signals by the corresponding participant's weight and comparing the model's performance.

Following this, the acceleration was filtered with a real-time 4th-order Butterworth filter. Both a low-pass filter and a band-pass filter were used and compared, with cut-off frequencies of 20 Hz and 0.1/20 Hz, respectively (16, 91).

Afterward, the variables of interest were selected from the dataset, namely the 3D acceleration, HR, GSR, and ground-truth MC. Furthermore, two other variables were computed from the 3D acceleration: its derivative and its vector norm/magnitude (Equation 2). These additional features were used to train the regression models both instead and in addition to the 3D acceleration, and their results were compared to the results achieved by using only the 3D acceleration. Along with the acceleration, HR, and GSR, the body mass index (BMI) and body mass of each participant were also given as features to the model. All these input variables were tested to verify their impact on the models' performance.

$$|v| = \sqrt{x^2 + y^2 + z^2} \quad (2)$$

Along with the removal of the variables unused in this work, some activities were also withdrawn from the datasets, and only the following were maintained: (i) standing; (ii) sitting; and (iii) walking. These activities were selected due to their resemblance to the industrial activities with a higher risk of WMSDs, presented in Chapter 3. This was followed by the creation of three separate sub-datasets. These sub-datasets were used to train three different regression models (activity-specific models) and their performances were compared to the complete dataset’s performances.

Subsequently, and repeating the strategy followed by Gjoreski et al. (17), the datasets were segmented in 10-second windows. For each window, to obtain the final feature vectors used to train the regression models, it was computed the average and the mean absolute deviation (MAD) of each variable, and both these methods were compared. The ground-truth MC of each window (i.e., the label) was obtained by computing its mean during that period of time. Additionally, the windows were tested with and without overlaps of 5 seconds (17), used to enable faster updates of the MC estimation in real-time. Figure 11 presents a diagram that better explains the process implemented to obtain the data that was used to train the regression models without (up) and with (down) 5-second overlaps.

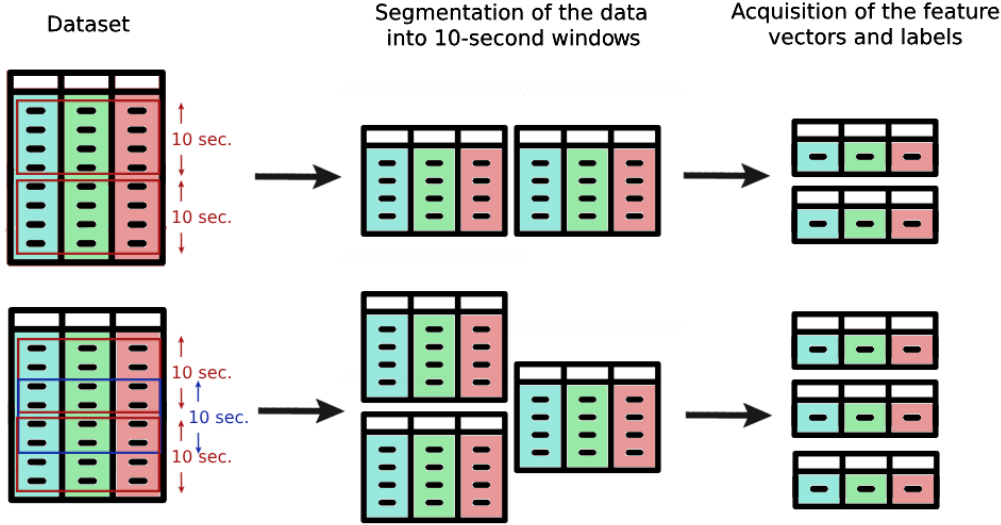


Figure 11: Diagram of the process of segmentation of the dataset into 10-second windows, with and without the 5-second overlaps (bottom and top diagrams, respectively).

After obtaining the features and labels for each participant the datasets were balanced to provide the same amount of data from each participant to the regression models. This was done by analyzing which participant had the least data, and then removing the excess data from the other participants to match the size of the smaller table.

Then, given the different sampling rates of each sensor, an interpolation method was applied

(piecewise cubic interpolation) to the HR and EE signals (18, 73). Additionally, the dataset was analyzed to verify the existence of outliers. By studying the average MC, and its standard deviation, for the walking, sitting, and standing sub-datasets, it was noticed that no participant had an average MC superior to the average of all participants by 2 times the standard deviation, and, therefore, no participant was removed from the dataset (14).

The final step was the normalization of every variable across all participants. Three different normalization methods were compared: median normalization, min-max normalization, and z-score normalization. Table 12 presents a synthesis of the various preprocessing steps tested, where ACC represents the 3D acceleration, ACC' and |ACC| its derivative and vector norm, respectively, and BW represents the body weight.

Table 12: Conditions tested during the data preprocessing. Legend: ACC - "Acceleration"; BW - "Body weight"

Order	Preprocessing Step	Conditions tested
1	HR and MC normalization by the participants' body mass	Yes/No
2	Acceleration filtering with 4th order Butterworth filter	Low-pass (20 Hz) Band-pass (0.1/20 Hz)
3	Input variables selection	ACC + HR + GSR + BMI ACC + HR + BW + BMI ACC + HR + BMI ACC + ACC' + HR + BMI ACC' + HR + BMI ACC + ACC + HR + BMI ACC + HR + BMI
4	Segmentation of data into the 10-second windows	5-second overlaps (Yes/No)
5	Obtention of the input features by computing a certain metric of the input variables for each 10-second window	Studied metric: Mean/MAD
6	Data normalization	Median/Max-Min/Z-score

4.2.3 Regression Models

Following the data preprocessing, several regression models (machine learning models and CNNs) were trained, validated, and tested. Before the models' training, the dataset and each sub-dataset were

divided into training/validation data and test data by randomly selecting a participant for testing the model. Then, the training data was shuffled. The validation method enforced was the LOOCV, a form of k -fold cross-validation where k (the number of folds) is equal to the number of subjects used for training. During k -fold cross-validation, the training dataset is divided into k sets, where $k-1$ subjects' data is used to train the model, and the other subject's data is used to validate it. This process is repeated k times, meaning that each subject was used for validation once.

CNNs with a regression layer

The CNNs were implemented using a team-owned deep-learning regression tool implemented in MATLAB (2022b, The Mathworks, Natick, MA, USA). Several models were trained, with different architectures and hyperparameters.

Each CNN was composed of one to three convolutional layers, each followed by a rectified linear activation function (ReLU) layer and an average pooling layer with a pool size and stride of 2. After the convolutional layer(s), a global average pooling layer was implemented, followed by two fully connected (FC) layers and, at last, a regression layer. The model's hyperparameters optimized for MC estimation were: (i) the number and size of the filters on each convolutional layer, (ii) the number of hidden neurons on the first FC layer, (iii) the learning rate, and (iv) the batch size. Table 13 summarizes the hyperparameters studied for the CNN, presenting the tested values for each parameter. Some hyperparameters' are represented by a range of values. In these cases, various values in the range were verified, but not all.

Table 13: *Tested hyperparameters for the CNN*

Hyperparameter	Values
Number of convolutional layers	1, 2, or 3
Number of filters	8 to 360
Filter size	5 to 30
Hidden neurons	0 to 1000
Learning rate	0.01, 0.005, or 0.001
Batch size	8 to 360

Regression Learner APP

The machine learning models were implemented using the Regression Learner APP, a MATLAB (2022b, The Mathworks, Natick, MA, USA) tool that allows a simple and straightforward training, validation, testing, and optimization of various regression models. The regression models used varied from: (i) LRs; (ii) decision trees; (iii) SVMs; (iv) GSVMs; (v) GPRs; (vi) kernel approximations; (vii) tree ensembles; and (viii) NNs. Initially, the hyperparameters used for each model were the ones predetermined by the Regression Learner APP, then, the best model's hyperparameters were optimized using three different techniques: (i) BO; (ii) grid search; and (iii) random search. Table 14 presents an overall view of all the implemented machine learning models in the Regression Learner APP.

Table 14: Machine learning models implemented in the Regression Learner App

Model family	Type	Hyperparameters	Model family	Type	Hyperparameters
LR	Simple	Max steps: 1000	GPR	Matern 5/2	Sigma: automatic
	Stepwise			Exponential	
Decision tree	Fine	Min. leaf size: 4	Kernel approximation	SVM	Kernel scale: automatic
	Medium	Min. leaf size: 12		Least squares	
	Coarse	Min. leaf size: 36	Tree ensemble	Boosted	Min. leaf size: 8
SVM	Linear	Kernel scale: automatic		Bagged	Min. leaf size: 30
	Quadratic		NN	Narrow	Neurons: 10
	Cubic			Medium	Neurons: 25
GSVM	Fine	Kernel scale: 0.9		Wide	Neurons: 100
	Medium	Kernel scale: 3.6	Bilayered	Neurons: 10/10	
	Cubic	Kernel scale: 14	Trilayered	Neurons: 10/10/10	
GPR	Rational Quadratic	Sigma: automatic			
	Squared Exponential				

In terms of the LRs, two different strategies were studied. The first model was a simple LR and the second regression model was a stepwise LR, with a maximum number of steps of 1000. Regarding the decision trees, three types of models were tested: a fine tree, a medium tree, and a coarse tree, with a minimum leaf size of 4, 12, and 36, respectively.

As to the SVM models, three regressors were analyzed, varying from each other on the kernel functions implemented (linear, quadratic, and cubic kernels). Additionally, three types of GSVMs were also studied, namely a fine, medium, and coarse model, with manual kernel scales of 0.9, 3.6, and 14, respectively.

Regarding the GPRs, four different models, with different kernel modes, were studied: a rational quadratic GPR, a squared exponential GPR, a matern 5/2 GPR, and an exponential Gaussian process

regressor (EGPR). As to the kernel approximation models, two learners were analyzed, an SVM kernel and a least squares kernel. Furthermore, two types of ensembles of decision trees were trained, a BDTR and a bagged decision tree, both with a minimum leaf size of 8 and 30 learners.

Lastly, five NNs were also analyzed. Three of those NNs had one single layer, each one with a different number of neurons. The narrow NN had 10 neurons, the medium NN had 25, and the wide NN had 100 neurons. A bilayered and trilayered NN, with 10 neurons on each layer, were also tested.

4.2.4 Model integration in SmartOs' architecture

The best-performing model, which was first developed, trained, validated, and tested using MATLAB, was then implemented in the high-level control system of the exoskeleton, on the UDOO board. This conversion was performed using MATLAB Coder APP/Codegen, a program used to generate C/C++ code from the MATLAB code.

To generate the C++ code, firstly, a function that used the regression model to predict the MC from one feature vector was created in MATLAB, this was the Coder APP's entry-point function. Then, an example code that used the entry-point function was created for the Coder APP to determine the type of variables used by the function. Finally, the Coder APP generated a standalone C++ code that replicated the entry-point function. The code generated by the Coder APP was then integrated into the exoskeleton's control architecture.

Then, following the general structure of SmartOs' gait analysis algorithms, the code generated was adapted and condensed to fit the existing architecture. Figure 12 presents the algorithm of the high-level controller.

The algorithm presented was integrated into the CCU of the system, which communicates with the WMSS board (acquisition of 3D acceleration data from InertiaLab's sensors), the LLOS board (mid- and low-level control of the exoskeleton), and the mobile APP that configures the system and begins the assistance. The regression model requires the accelerometers' data from the ankle, waist, chest, and wrist, obtained by the WMSS system (InertiaLab), and the persons' anthropometric data, namely their height and weight used to compute the BMI, which are provided to the mobile APP and transmitted to the CCU.

Regarding the code developed in the WMSS board, presented in the top block of Figure 12, its purpose was to send the 3D acceleration data, measured by the four IMUs (InertiaLab's sensors), to SmartOs' CCU. Every 10 milliseconds, the board reads the acceleration along each axis, processes this data according to the initial sensors' calibration, and sends it to the CCU.

In regards to the code developed in the CCU board, presented in the bottom block of Figure 12, its

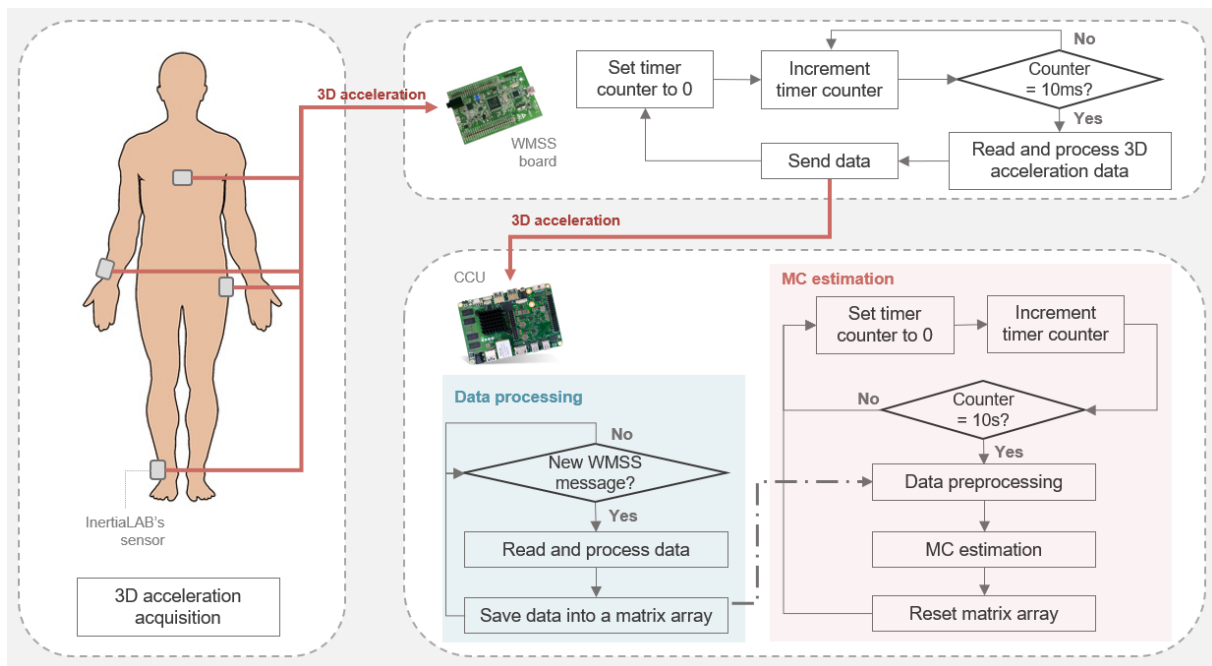


Figure 12: Fluxogram depicting the code developed in SmartOs' architecture that allowed for the MC estimation in real-time based on the data from 4 accelerometers.

purpose was to read the 3D acceleration data provided by the WMSS board and use this data to estimate the person's MC in real-time. The code was divided into two main blocks - data processing (blue block in Figure 12) and the MC estimation (pink block in Figure 12). The data processing block was established to make the CCU read, process, and save the 3D acceleration data after a message from the WMSS data was received. The MC estimation block was executed every 10 seconds, something made possible by a timer, and was composed of the necessary algorithms to estimate the MC based on the 3D acceleration of the chest, left waist, right wrist, and right ankle measured in the past 10 seconds and the person's BMI.

The IMUs orientation of Ingraham's dataset was studied to use the same axis orientations when using InertiaLab's IMUs. This was done by conducting a walking test, on a treadmill, during which the InertiaLab's IMUs were used to obtain the 3D acceleration of the chest, right wrist, left waist, and right ankle, at a speed of 3 km/h, and comparing the acquired data to Ingraham's acceleration data respective to walking at the speed of 3.2 km/h. The comparison was done by studying the cross-correlation of the different signals for the four pairs of IMUs (Ingraham's and InertiaLab's) using MATLAB's tools. The cross-correlation results are presented in Appendix A. Afterward, it was performed a visual inspection of the achieved results where the similarity between the signals with higher correlation was analyzed and the axis correspondence was verified. For this purpose, it was analyzed if the rotation of the axis from Ingraham's configuration to the InertiaLab's configuration would be possible by following the right-hand rule. Figure 13 presents the axis correspondence between the two IMUs sets.

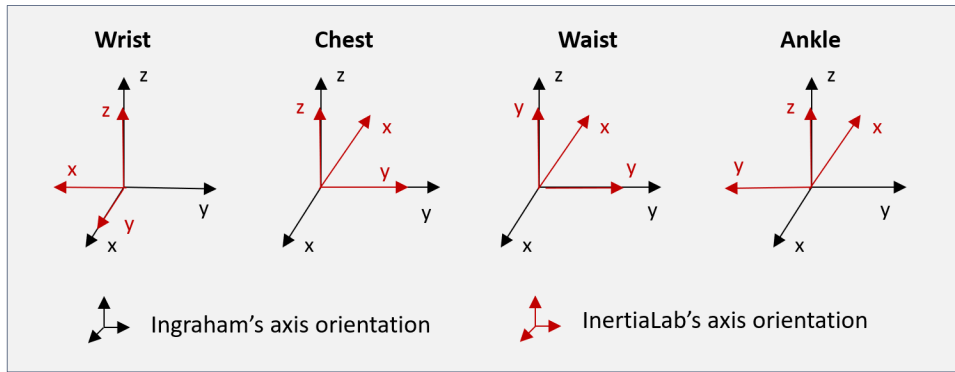


Figure 13: Representation of InertiaLab's axis orientation in relation to Ingraham's axis orientation.

Lastly, the SmartOs' mobile APP was adjusted. To enable the activation of the MC estimation through the APP an additional setting was added to the system's configuration interface. This step was performed using Android Studio's IDE. The regression model needs the data obtained by the InertiaLab's sensors (IMUs), therefore, when the "MC Estimation" option is toggled, the communication between the InertiaLab's sensors and the CCU, through the WMSS board, is turned on by forcefully toggling the "Inertial Lab" checkmark. Figure 14 shows the changes performed to the APP.

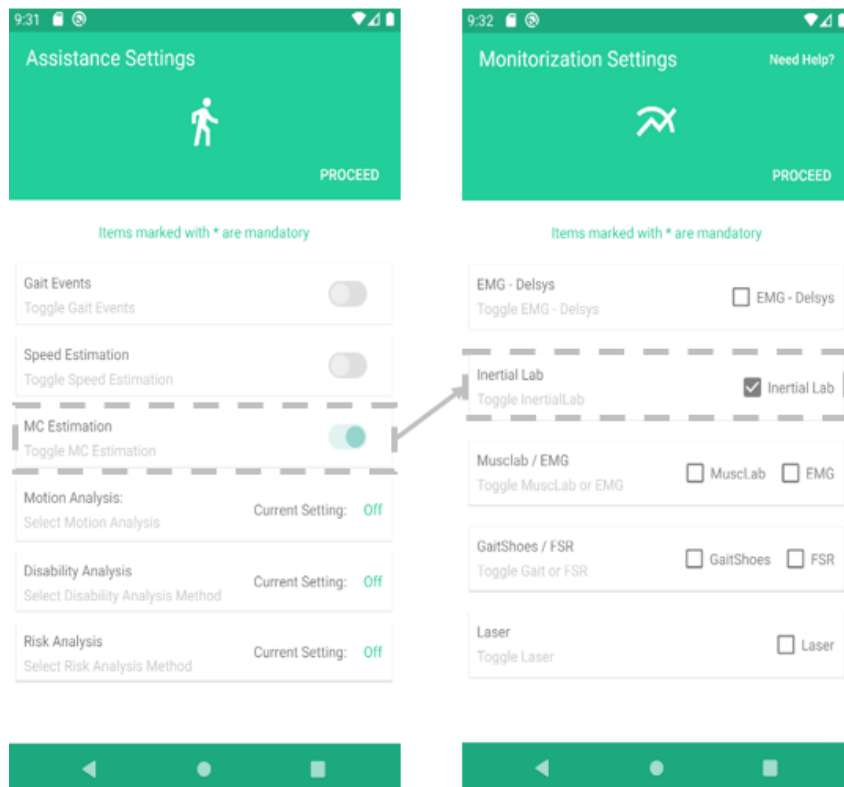


Figure 14: Interface options added to SmartOs' APP that enabled the start of MC estimation.

4.3 Validation

This section presents the validation of the developed regression model used for MC estimation. The validation phase was branched into three stages: (i) offline phase, used to optimize the preprocessing method, find the best model's inputs, and study the best-performing model; (ii) bench tests, performed to access the correct functioning of the best-performing regression model; and (iii) human experiments, to validate the regressor in real-time, when compared to the ground-truth.

4.3.1 Offline

The offline validation was performed to obtain the best possible performance when estimating the MC, by using Ingraham's dataset (66) to train and validate different regression models through the LLOCV method. Various aspects of the regression model training were analyzed and optimized, namely: (i) the data preprocessing method; (ii) the model's inputs; (iii) the use of activity-specific models; (iv) the best models' hyperparameters; and (v) the best regression algorithm.

The various studied models were evaluated regarding the metric obtained during the validation process, namely the RMSE, which measures the difference between the predicted values and the target values (ground-truth of the MC, measured through indirect calorimetry), and the coefficient of determination (R^2), which assesses the fit quality of the regression models. The performance of each regression model was obtained for each of the validation iterations of the LOOCV method, and, in the end, the average and standard deviation of each metric were computed. The RMSE was computed using Equation 3, where N is the number of predictions, $y(i)$ is the i^{th} ground-truth MC, and $\hat{y}(i)$ the i^{th} MC estimated by the EGPR model.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (y(i) - \hat{y}(i))^2}{N}} \quad (3)$$

4.3.2 Bench tests

The following bench tests were performed to validate the performance of the regression model for real-time MC estimation. Firstly, the regression model implemented in C++ was compared with the original model developed in MATLAB. This was done by testing the regression model with the input data from the test participant. The MC prediction made by the C++ model was then compared to the ground-truth MC and the MATLAB prediction.

Then, the computational load of the regression model was analyzed. This validation process was done by analyzing the time taken to execute each function of the MC estimation algorithm. To do this, an output pin of the CCU board was activated every time each function was called and deactivated when it ended. The timings of the algorithm were then obtained by inspecting this pin in an oscilloscope. The results were organized in a time diagram to verify any time constraints in the code.

4.3.3 Human experiments

The experimental validation of the regression model model was performed to verify the accuracy of its MC estimation. In this phase, the ground truth of the MC was obtained through indirect calorimetry, by equipping participants with a respirometer device (K5, COSMED, Italy). The participants also wore the four IMUs necessary for estimating the MC and performed the three motor activities studied: standing, walking, and sitting. The prediction made by the regression model was then compared to the MC estimated through indirect calorimetry.

Additionally, participants were also equipped with motion track markers (Qualysis, Sweden) in their lower limbs, a chest HR monitor (HRM Dual, Garmin, U.S.A), and three additional InertiaLab IMUs placed on the right knee, left waist, and back side of the waist. The motion track system was used to study the kinematics and kinetics of the lower limbs during walking. The participants walked on a force plate-instrumented treadmill (AMTI, U.S.A) that measured the bilateral forces during the gait. The protocol was performed in two days, at Porto Biomechanics Laboratory (LABIOMEPE), University of Porto.

Participants

Five volunteers (3 males and 2 females) participated in this study after giving their informed consent. The participants were healthy individuals with no history or evidence of locomotor or balance impairments and did not suffer any musculoskeletal injury six months prior to this experiment. The participants, whose demographic information is presented in Table 15, had ages between 22 and 29 years old (25 ± 2.9 yr), body masses between 65 kg and 99 kg (77.8 ± 13.4 kg), and BMIs between 24.0 kg/m² and 29.2 kg/m² (26.8 ± 2.0 kg/m²). This protocol was conducted under the ethical procedures of the Ethics Committee in Life and Health Sciences (CEICVS 006/2020), following the Helsinki Declaration and the Oviedo Convention.

In contrast to Ingraham's protocol (66), this study was more balanced regarding its participants' gender, with 40% of the participants being females, against the 20% in Ingraham's dataset. Additionally, two participants' BMIs were out of the Ingraham dataset's BMI range (18.6 to 27.8 m/kg²), Participant

Table 15: Validation participants' demographics

Participant	Age (years)	Gender	Weight (kg)	Height (m)	BMI (kg/m²)
1	29	M	76	1.78	24.0
2	24	M	99	1.84	29.2
3	23	F	68	1.60	26.6
4	27	M	81	1.70	28.0
5	22	F	65	1.58	26.0

2 was heavier than the heaviest Ingraham's participant (95.2 kg), and Participants 3 and 5 were smaller than the smallest Ingraham's participant (1.63 m). Furthermore, two of the participants (3 and 5) were younger than the youngest Ingraham's participant (24 yr).

Experimental protocol

The participants were first equipped with seven InertiaLab IMUs in the right ankle, right knee, right wrist, right waist, left waist, back waist, and chest. These sensors were connected to the WMSS board, which was linked to the CCU of the SmartOs' system. The WMSS board, the CCU, the system's battery, and the power supply interface were all placed on a backpack worn by the participants. Afterward, the participants were equipped with a chest HR monitor.

Then, the motion track markers were placed on the participants' lower bodies. Figure 15 presents each marker's location. A total of 44 markers were placed, 8 of which were placed on the hip area, 8 on the upper legs, 8 on the knees, 8 on the lower legs, and 12 on the feet.



Figure 15: Locations of the lower body motion track markers.

Afterward, the participants were equipped with the respirometer. The face mask was tightened enough to prevent air from escaping the system but not too tight to make its use too uncomfortable. The respirometer backpack was also placed on the participants' backs, above SmartOs' backpack. Figure 16 presents the sensors' placements when worn by a participant.

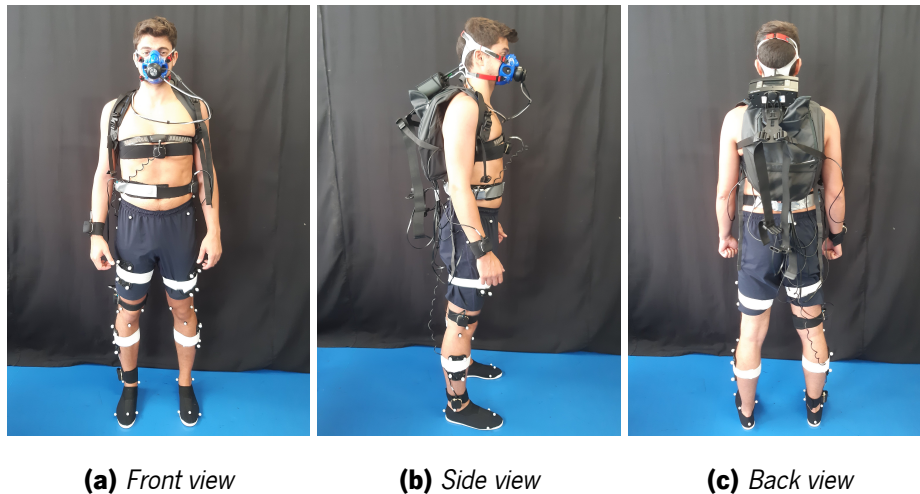


Figure 16: Participant instrumented with the wearable sensors, namely, the InertiaLab's IMUS, motion track markers, HR chest monitor, and the respirometer device.

The data acquisition was then started. Firstly, the participants stood completely still for 5 seconds to calibrate the IMUs. Afterward, a static trial was performed to calibrate the motion track software, with the participants also standing still. After the calibration was concluded, the MC estimation was turned on simultaneously with the respirometry acquisition.

The participants performed 5 motor tasks: (i) standing; (ii) walking at 1.5 km/h; (iii) walking at 2 km/h; (iv) walking at 3 km/h; and (v) sitting. Each motor task lasted 10 minutes and the participants did not rest between tasks. The order of the different walking tasks (ii, iii, and iv) was randomized. The standing and walking activities were performed on the instrumented treadmill, and the sitting activity was performed on a chair right next to the treadmill, meaning that the participants had to move between the last walking task and the sitting task. Figure 17 presents the three types of activities performed by the participants. In Figure 17, it is also possible to observe the instrumented treadmill and the chair used during the experiment.

It was asked to the participants to be as still as possible during the standing and sitting tasks and to not rest their right arm on the treadmill. The motion capture system was only turned on for one minute at the end of each walking task. The treadmill was composed of two force-sensing plates with a front-to-back split configuration, and, during the motion capture period, the participants were asked to only place one



(a) Standing

(b) Walking

(c) Sitting

Figure 17: *Activities performed by the participants.*

foot on the front plate at a time, after the other foot was entirely on the back plate.

Due to a sudden stop in SmartOs' system halfway through Participant 2 acquisition, the participant was asked to repeat the protocol on the second day, resulting in the execution of two different trials by the participant. Additionally, the SmartOs' system also failed midway through the protocols of Participants 2 (during their second trial) and 5.

Data collection and analysis

The data acquired during the protocol comprised: (i) the 3D acceleration of the right ankle, right knee, right waist, back waist, left waist, chest, and right wrist, measured by InertiaLab's sensors and stored by the SmartOs system; (ii) the HR, measured by a chest monitor; (iii) the rate of oxygen consumption and carbon dioxide production, measured by a respirometer device; (iv) the gait kinetics and kinematics of the lower body, measured by motion capture markers and a force-instrumented treadmill; (v) the estimation of the MC made by the SmartOs system. The data acquired with the optical markers and the force plate-instrumented treadmill did not concern the work presented in this dissertation and thus are not analyzed in this manuscript.

Figure 18 is composed of four graphics depicting the 3D acceleration of Participant 4, measured by the IMUs placed on the chest, right wrist, left waist, and right ankle, which were the signals used by the regression model to estimate the MC. The 3D acceleration of the remainder participants is presented in Appendix B.

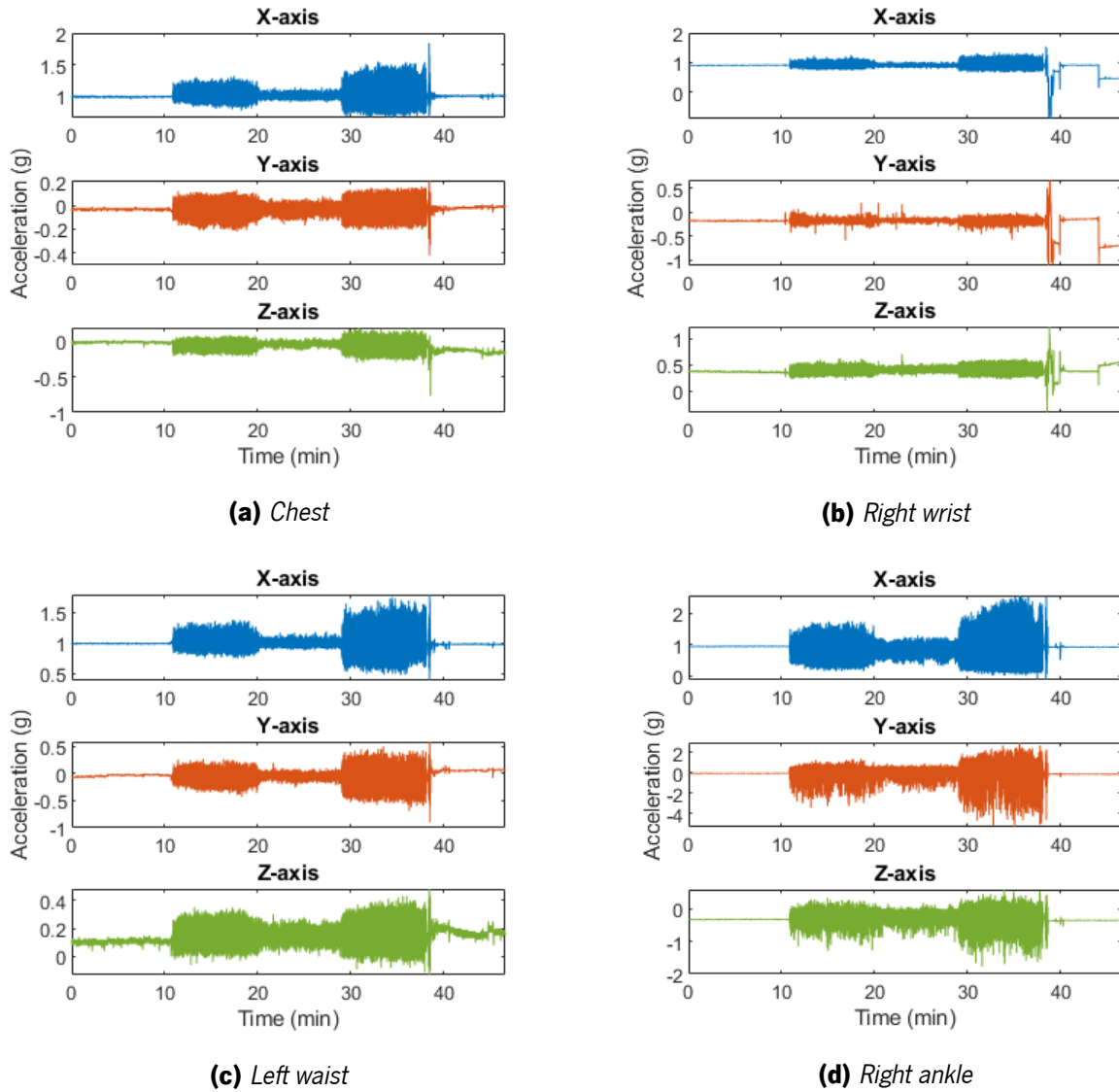


Figure 18: Raw 3D acceleration signals measured by the InertiaLab’s IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 4.

Figure 19 presents the HR of the five participants (including both trials of Participant 2) during their respective protocols. Each participant’s plot is labeled according to the activities being performed at each time. Additionally, the average HR of each participant is presented at the top of each plot.

For validating the real-time performance of the regression model, the SmartOs data and the respirometer data were processed and compared in MATLAB (2022b, The Mathworks, Natick, MA, U.S.A). Firstly the relevant data was selected from the files, i.e., the acceleration and MC estimation, from the SmartOs’ files, and the rate of oxygen consumption ($\dot{V}O_2$), carbon dioxide production ($\dot{V}CO_2$), and HR from the respirometer data. Two separate data structures were created - one for each system. The structures were processed to present the acquisition time in minutes and an activity label to

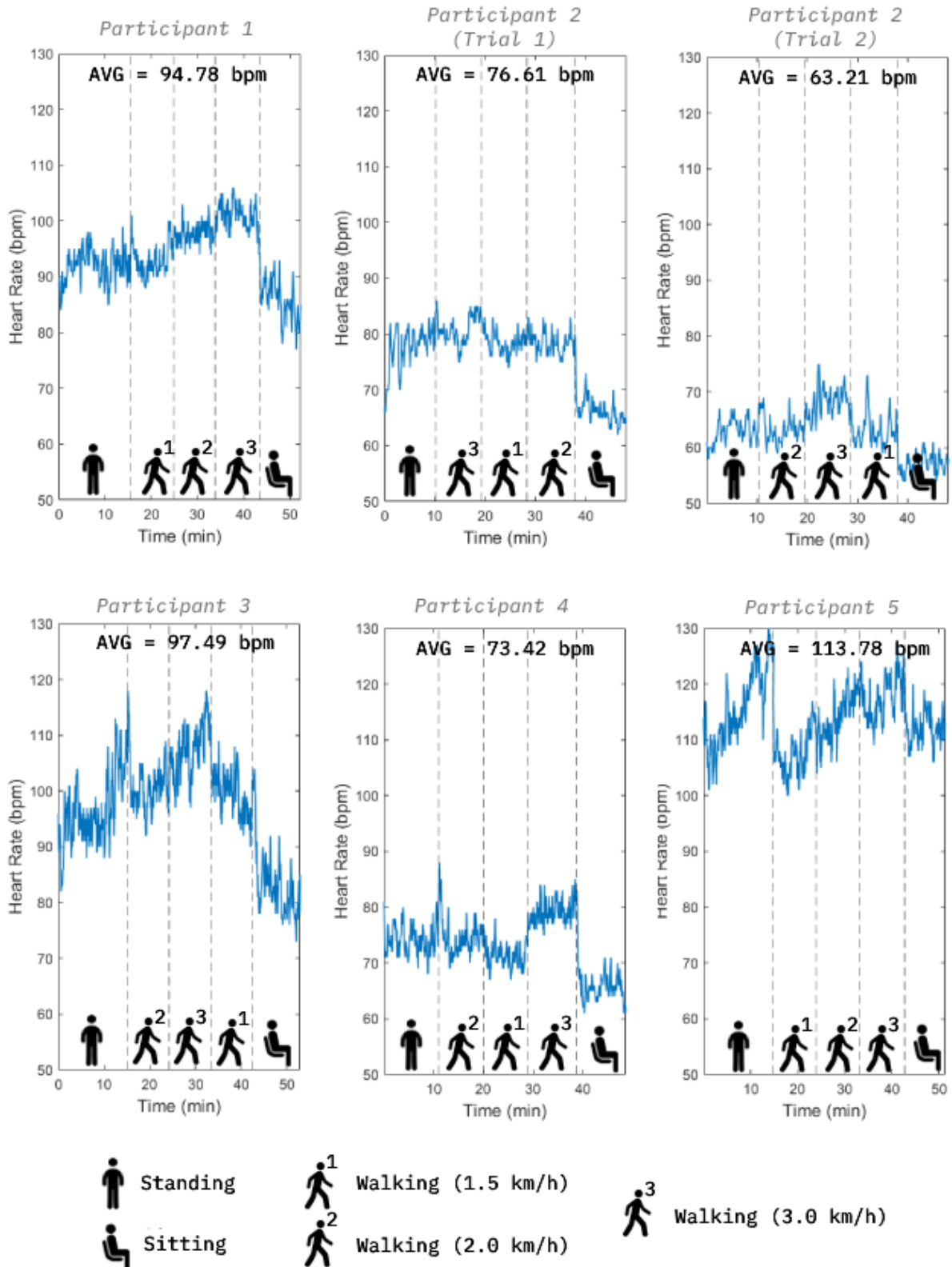


Figure 19: Heart rate of each participant during the experiment, and respective average.

identify the performed task at any time period. Afterward, the first three minutes of respirometer data were removed from the structures, since the respirometry data only achieves a steady-state after that time period.

The ground-truth from the respirometer data structure was obtained following three different approaches: (i) the per-breath respirometry; (ii) the steady-state respirometry; and (iii) the fast-estimated respirometry. This was based on the approach followed by Slade et al. (15), which compared a model for MC estimation with the results achieved by these three indirect calorimetry methods. The per-breath respirometry was obtained by applying Brockway's equation (Equation 1) to the respirometer data measured at every breath. This MC was normalized by the weight of each participant and averaged into 10-second periods to match the estimation frequency of the regression model. The steady-state respirometry was obtained by averaging the MC, also estimated by Brockways' equation, during the steady-state period. The fast-estimated respirometry was computed by fitting a first-order dynamical model to the per-breath MC data (obtained with Brockway's equation as well) acquired during the steady-state phase. A First-order dynamical model was chosen for comparison since it is the most common method to estimate the MC in HITL controllers, as shown in Chapter 2.2. The dynamical model used was presented and described by Zhang et al. (8).

4.4 Results and discussion

This section presents the results obtained during the three validation phases (offline, bench tests, and human experiments), as well as the results obtained by an offline test to the best-performing regression model, concluded after the offline validation. The offline validation results address the various optimization steps executed to obtain the ideal MC estimator. The bench test's results concern the model's performance in C++ and its computational load to the system. The human experiments' results refer to the real-time validation of the regression model, i.e., its comparison to the MC estimated by indirect calorimetry.

4.4.1 Offline validation

Preprocessing methods

Regarding the data preprocessing steps, the following aspects were studied: (i) the filtering of the acceleration signals, comparing the effects of low-pass and band-pass Butterworth filters; (ii) the effect of normalizing the HR and MC to the user's body mass; (iii) the effect of segmenting the 10-second windows in 5-second overlaps; (iv) the metric used to compute the final features on each window; and

(v) the final normalization method. These tests were conducted by using the same regression model for each experiment, and only changing the test conditions. Figure 20 presents an overview of the best preprocessing steps.

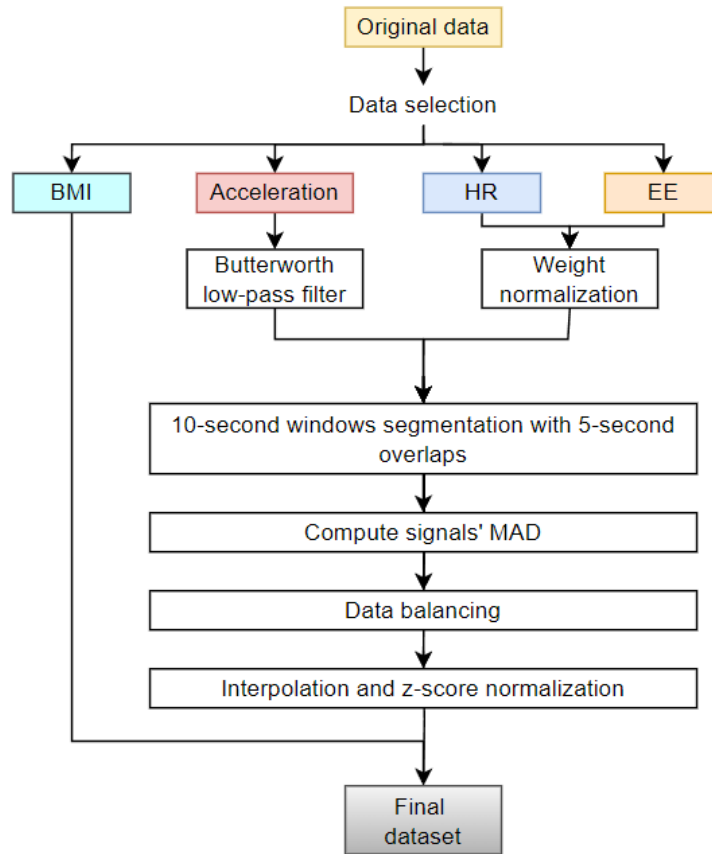


Figure 20: Diagram of the preprocessing steps that resulted in the best regression model performance.

The results showed that the preprocessing methods used to prepare the data can affect the efficacy of a regression model. Better results were achieved by using a low-pass filter, as performed by Su et al. (91), to remove high-frequency noise from the acceleration since the bandpass filter ended up erasing small variations in these signals that were related to changes in activities. The model also performed better when the HR and MC were normalized by the users' body mass, compared to using the body mass as an input estimator, demonstrating that a person's body mass impacts their MC. This normalization was also performed by two literature studies (16, 66).

Overlapping the 10-second feature windows every 5 seconds like Gjoreski et al. (17) also resulted in an increase in the model's performance due to an increment in the data given to the regression model. Giving the MAD of each signal (3D acceleration, GSR, and HR), instead of the mean of each signal, as performed by Bazuelo-Ruiz (92), also improved the MC estimation, since this metric was useful for identifying the distance of the measurements to the signals' means (i.e. the signals' variability). Therefore, the variability

of the signals proved to be more related to the MC measured. Additionally, the best normalization method turned out to be the z-score method, where the data is normalized based on its mean and standard deviation, which could possibly be explained by the effectiveness of this method in handling outlier data.

Model’s inputs

Several signals were given to the regression models during the training process, namely the 3D acceleration of the chest, right wrist, left waist, and right ankle, the derivative and vector norm of each acceleration signal, the HR, and the GSR. Furthermore, two anthropometric features were also tested as inputs: the body mass and BMI. Best validation results were achieved when using the following features to train the regression models: (i) the 3D acceleration on the four locations, (ii) the HR, and (iii) the BMI.

These results resonate with the results achieved in the literature that proved the 3D acceleration and HR to be highly related to the MC (66, 72). Additionally, it was possible to conclude that providing more inputs to the models did not always improve their performance, since some estimator signals, namely the acceleration vector norm, acceleration derivative, and GSR did not correlate as much to changes in the participants’ MC. Using these signals, therefore, resulted in biased models, i.e., models less capable of generalization. Using the body mass as an input signal did not improve the performance likely due to the weight normalization previously performed during the preprocessing phase.

However, for the HITL application, it is desired to reduce the number of sensors as much as possible. Therefore, the effect of using only the 3D acceleration and BMI data to train the EGPR model was studied. Table 16 presents the performance of the compared models: (i) a model that used the HR, 3D acceleration, and BMI; and (ii) a model that used the 3D acceleration and the BMI.

Table 16: *Performance comparison between the model that was trained with the HR, acceleration, and BMI, and the model that was trained with the acceleration and BMI*

	Model trained with the HR, acceleration, and BMI	Model trained with the acceleration and BMI
Validation RMSE (W/kg)	0.304 ±0.014	0.308 ±0.026
Test RMSE (W/kg)	0.412	0.45

Despite the validation error increasing by 0.004 W/kg, the test error increased by 0.038 W/kg when the HR signal was removed from the training data. However, this increase was not considered to be significant, as the regression model was still capable of accurately estimating the MC based only on the four 3D acceleration signals. The regression model presented here was, therefore, more feasible for

HITL applications than most models used in the literature and could lead to less obtrusive exoskeleton assistance for workers during occupational tasks. Consequently, the input estimators more adequate for HITL applications are: the 3D acceleration of the chest, left waist, right wrist, and right ankle, and the BMI.

General vs activity-specific models

Table 17 presents the comparison of the results when using activity-specific regression models (different regression models trained for different activities with the sub-datasets) to the performance of a global regression model (trained with the full dataset), when using a CNN with the same architecture. The results showed that using the activity-specific regression models led to a slightly better RMSE during validation, but worse R^2 compared to the general regression model. However, when predicting the MC of the test subject, the activity-specific models significantly decreased their performance in both RMSE and R^2 metrics.

Table 17: Validation and test performance of the three activity-specific models (walking, sitting, and standing) and the general model

	Walking	Sitting	Standing	All activities
RMSE (Val)	0.42	0.39	0.26	0.5
R² (Val)	0.49	0.46	0.33	0.73
RMSE (Test)	0.57	1.13	0.4	0.41
R² (Test)	0.08	-4.69	-1.73	0.84

The results showed that the activity-specific models suffered from overfitting. This was believed to be due to the smaller datasets used to train the activity-specific models, which resulted in an inaccuracy of the models when predicting the MC based on unseen data. Therefore, a general model was used in this work to estimate the MC of humans performing different motor tasks.

Best CNN

The CNN that achieved the best results during validation was a network with two convolutional layers, with 32 filters with a size of 10. Each convolutional layer was followed by a ReLU layer as the activation function, which was followed by an average pooling layer with a pool size of 2, stride of 1, and padding equal to the input vector's size. Then, the data went through a global average pooling layer, two fully connected layers with an output size of 50 and 1, in that order, and a final regression layer.

The batch size, number of epochs, and learning rate that led to the best performance were 32, 100, and 0.01, respectively. Additionally, it was used an L2 regularization method to reduce overfitting. The regularization term that resulted in the best validation results was $1e-4$. Figure 21 presents a diagram of the CNN's layers.

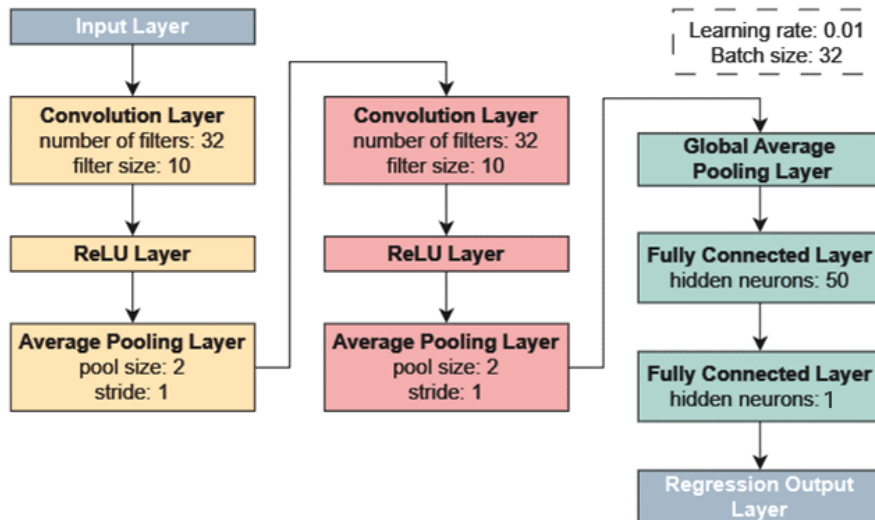


Figure 21: Layer diagram of the best performing CNN model.

These results showed that highly complex models are not necessarily more accurate in predicting the MC, since the best-performing CNN was a relatively simple NN, with only two convolution layers purely composed of 32 small filters, and only 50 hidden layers in the first fully connected layer. This revealed that there was no necessity for many layers to find patterns in the data.

Models' comparison

Table 18 presents the performance of the various machine and deep-learning models trained, including the validation results obtained by the best CNN. In total, 25 regression models were trained and validated.

These results showed that the model with the best performance (lowest RMSE and higher R^2) was the EGPR. The EGPR model's results were very similar to the rational quadratic GPR's. The third and fourth best models were also GPR models: the matern 5/2 and the squared exponential GPRs, respectively. This demonstrated the superiority of the GPR models in predicting the MC.

Additionally, despite being the most complex model in the study, the CNN was only the 10th best regression model during validation. This could be explained by the small size of the dataset used to train the models, as simple machine-learning models are generally superior when trained with less data. The

Table 18: Validation RMSE and R^2 of each regression model trained

Model	RMSE	R^2	Model	RMSE	R^2
LR	0.432	0.80	Matern 5/2 GPR	0.319	0.89
Stepwise LR	0.379	0.85	EGPR	0.306	0.91
Fine tree	0.400	0.83	SVM Kernel	0.422	0.81
Medium tree	0.371	0.85	Least squares Kernel	0.414	0.82
Coarse tree	0.360	0.86	BDTR	0.372	0.85
Linear SVM	0.435	0.80	Bagged decision tree	0.326	0.89
Quadratic SVM	0.379	0.85	Narrow NN	0.377	0.85
Cubic SVM	4.234	-18.3	Medium NN	0.397	0.83
Fine GSVM	0.342	0.87	Wide NN	0.396	0.83
Medium GSVM	0.358	0.86	Bilayered NN	0.347	0.87
Coarse GSVM	0.414	0.82	Trilayered NN	0.366	0.86
Rational quadratic GPR	0.306	0.90	CNN	0.366	0.82
Squared exponential GPR	0.331	0.88			

data previously presented in Figure 18 of Chapter 4.3.3 showed that the acceleration variation, along the three axes, was linearly related to the walking speed and that the transitions between the speeds could be noticed just by looking at the plots. This could explain why a simple machine-learning model achieves better results when estimating the MC based only on these signals.

The performance achieved by the EGPR model (RMSE of 0.31 W/kg) was better than the results achieved by the study performed on the same dataset (RMSE of 1,03 W/kg) despite using significantly fewer sensor data (66). It is important to note that Ingraham et al. used the model to predict the MC during heavier activities, namely running, cycling, and stair climbing, that are generally associated with lower MC estimation accuracy (66).

Optimization of EGPR's parameters

After observing that the model with the best performance was the EGPR, the Sigma parameter of the model was optimized by using three distinct methods: BO, grid search, and random search. The acquisition function used, i.e., the technique used to determine which hyperparameters are evaluated at each iteration, was the probability of improvement method. The results obtained by each optimization method are presented in Table 19, as well as the performance previously obtained when comparing all

the models by using MATLAB's Regressor Learner APP standard value for the Sigma parameter (0.1650).

Table 19: Validation RMSE of an EGPR model with a Sigma parameter optimized by different methods (BO, grid search, and random search) in comparison to no optimization process

	Optimal Sigma	RMSE (Val)
No optimization	0.1650	0.306 ± 0.018
BO	0.1492	0.304 ± 0.014
Grid search	0.2101	0.310 ± 0.164
Random search	0.1088	0.306 ± 0.017

Therefore, it was observed that the method that achieved the best results was the BO, however, there was not a significant difference in the validation RMSE obtained by the different techniques. Nonetheless, this optimization allowed for a reduction in the validation RMSE average of 0.002 W/kg.

4.4.2 Offline testing

The offline validation previously described resulted in the development of the best possible regression model for MC estimation. The optimized EGPR model was then used to predict the MC of the test subject based on the data acquired by the accelerometers and the participant's BMI. As previously mentioned, the test participant was chosen randomly. The RMSE and R^2 of the test prediction were 0.45 W/kg and 0.84, respectively. Figure 22 presents the prediction of the EGPR, in blue, in comparison to the target MC (ground-truth), in orange.

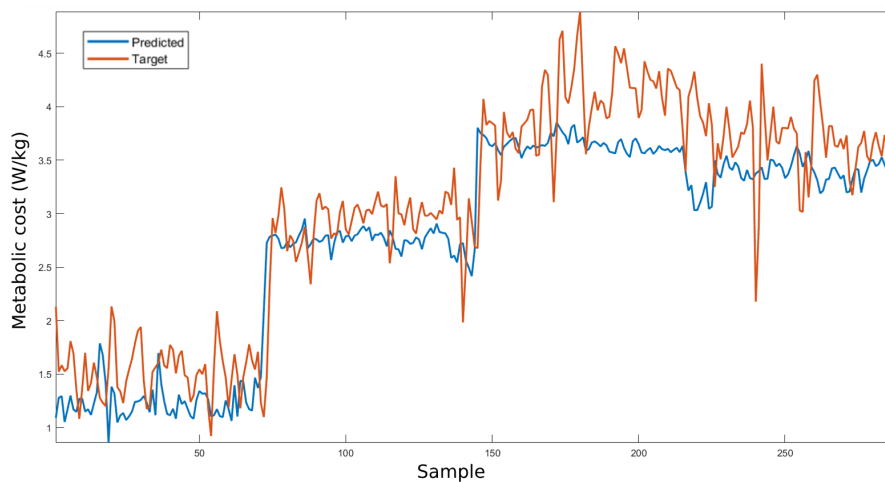


Figure 22: Comparison between the test subject's true MC (orange) and the estimated MC (blue).

The results revealed a slightly higher RMSE and smaller R^2 when compared to the performance during validation, with the RMSE increasing by 0.142 W/kg and the R^2 decreasing by 0.07. However, these results were satisfactory as the EGPR model achieved similar performance to the best model in the literature (0.36 W/kg (16)) despite using fewer input signals and being a much simpler regression model.

4.4.3 Bench tests

Accuracy of the MC prediction in SmartOs system

The EGPR model was implemented in the SmartOs system and its performance was compared with the one obtained in Matlab for the same data. It was observed that when given the same feature vector, the output (i.e., the MC estimation) was the same for both EGPR models (RMSE of $6.94e^{-6}$). Figure 23 shows the differences between the models' predictions, revealing the resemblance between the two curves.

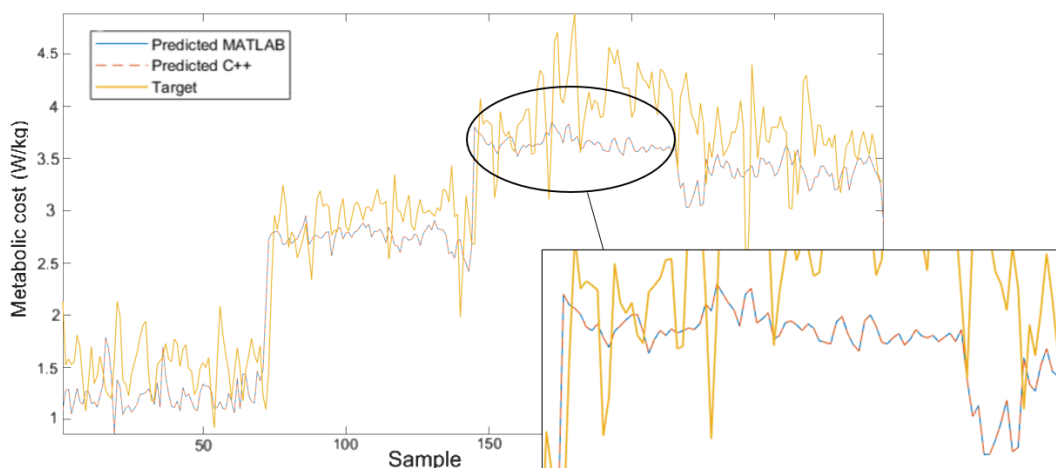


Figure 23: Comparison between the test subject's true MC (yellow), the estimated MC in MATLAB (blue), and the estimated MC in C++ (orange).

These results are ideal since they showed that the model developed in C++ was able to predict the MC with the same accuracy as the original model. Therefore, the code generated by the Coder APP was viable to be integrated into SmartOs' architecture.

Code timing analysis

Figure 24 presents the time diagram of the MC estimation algorithm, integrated into the SmartOs' CCU. The duration of each main function of the algorithm is presented in the diagram, namely the time needed to read and save the acceleration data into 10-second windows, to preprocess that data, and to estimate the MC.

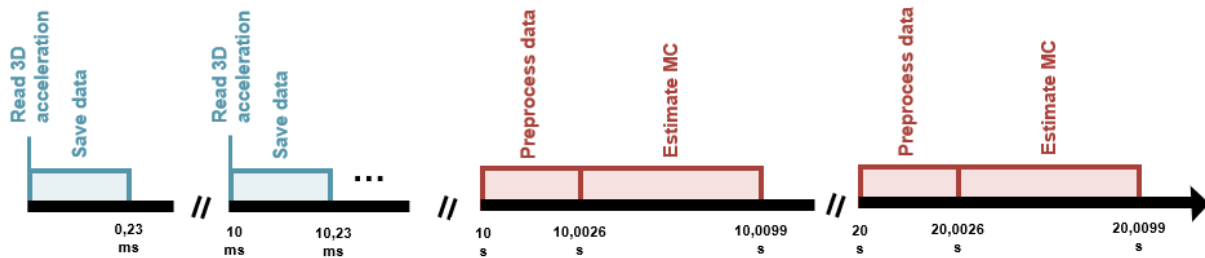


Figure 24: Time diagram depicting the time it takes to execute the different MC estimation algorithm's functions.

The diagram showed that saving the acceleration data happened, as expected, every 10 milliseconds since that was the frequency at which the WMSS board was programmed to send the data to the CCU. This process took, on average, 0.23 milliseconds. Therefore, this function did not disturb the communication between the two boards, which happens every 10 milliseconds.

The additional functions of the MC estimation algorithm were programmed to execute every 10 seconds. The results showed that this was successfully achieved. Additionally, the time diagram revealed that the data preprocessing took, on average, 2.6 milliseconds, and, the MC estimation took 7.3 milliseconds. The all process lasted, therefore, for 9.9 milliseconds.

Together, the three main functions took an average of 10.13 milliseconds, more than the 10 milliseconds necessary to undisturbingly read the data from the WMSS board. However, since the processes were developed into different threads (depicted using the colors blue and red) they were able to be executed simultaneously with no delays.

4.4.4 Human experiments

Figure 25 depicts the MC estimated, in real-time, by the EGPR model in comparison to the MC calculated by the three methods of indirect calorimetry, for the five participants, including the two trials performed by Participant 2. Furthermore, the activity transitions are also marked and labeled.

The prediction results presented in Figure 25 show that the accuracy of the prediction varied across the different trials. Additionally, it was possible to observe that the per-breath respirometry estimation was very noisy. The fast-estimated respirometry estimation was very close to the average of the MC for each activity (the steady-state respirometry estimation). In general, it was possible to see that the EGPR model underestimated the MC when compared to the ground truth in every trial, and this underestimation was higher for the participants who spent more energy during the protocol.

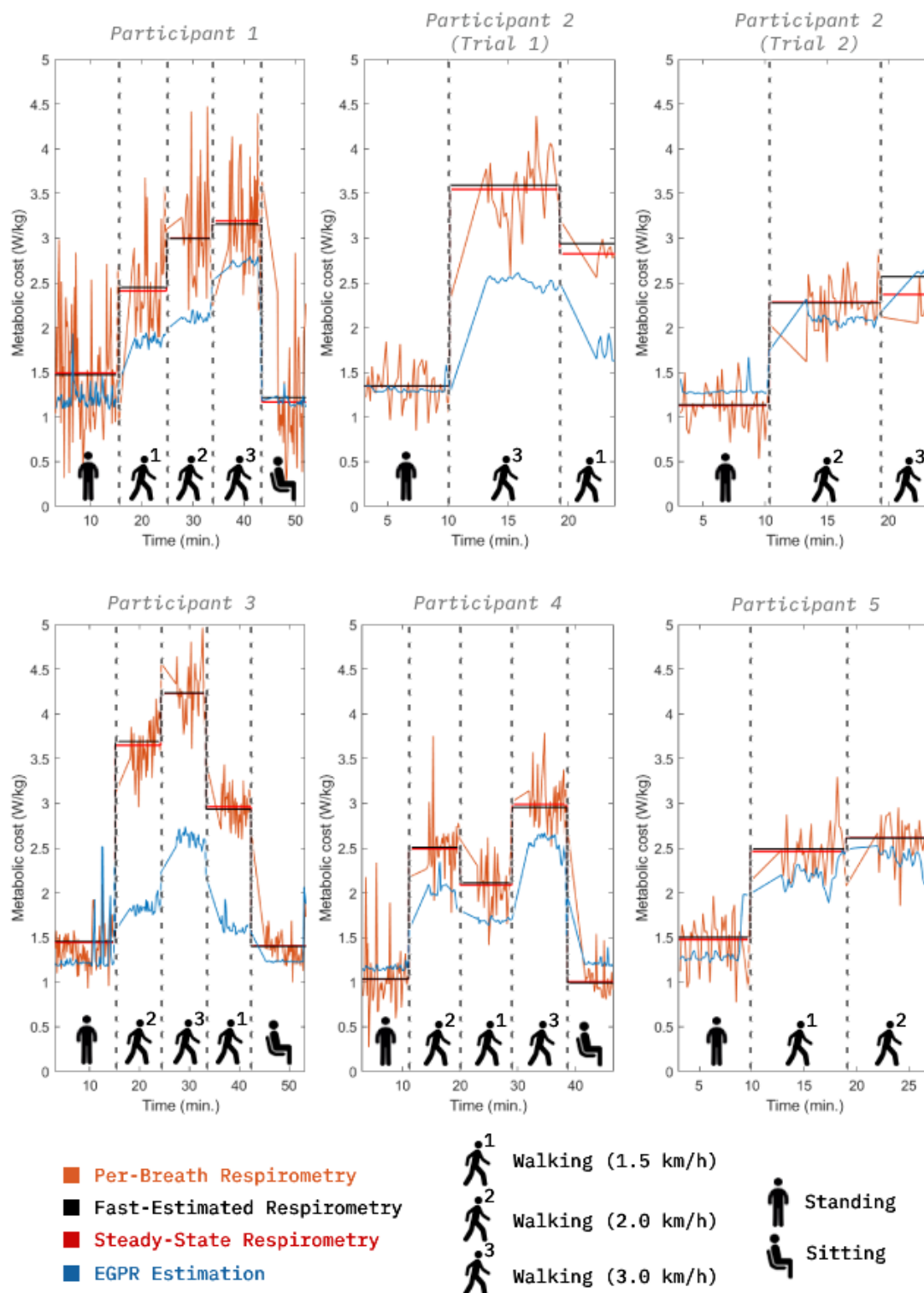


Figure 25: Comparison between the MC estimated by the three methods of respirometry (per-breath, fast-estimated, and steady-state, in orange, black, and red, respectively) and the MC estimated by the EGPR model (in blue), for each participant.

As seen in Figure 25, the difference between the regression model estimation and the respirometry estimations was significantly higher for Participant 3. The MC estimated by the model for this participant was similar to the MC estimated for the other participants, since the 3D acceleration was similar for all of them, however, the third participant had an abnormally high MC during the walking activities. Participant 3 presented a normal HR during the trial, close to the average HR for all participants, as seen in Figure 19, and had similar characteristics to Participant 5, regarding their age, gender, body mass, and height. Additionally, Participants 3 and 5 performed their respective trials at a similar time of the day, with similar room temperatures, and it is possible to observe that both participants had similar steady-state MC during the standing activity. Therefore, Participant 3's high MC could not be explained by any of the mentioned factors.

Additionally, it was possible to observe that the Participant 2 MC was higher during the first trial. One factor that varied between the two trials was the time of day: the first trial was performed after lunch, and the second trial before lunch. This could explain this difference since the basal MC naturally increases after a meal. This difference could also be related to the HR disparity noticed in Figure 19.











The closest prediction to the ground truth was obtained by Participants 2 (second trial) and 5, despite the participants presenting very different demographics. Additionally, both participants had different HR averages (63.3 bpm and 113.4 bpm for Participants 2 and 5, respectively), and the trials were performed at different times of the day. The remainder of this chapter presents some performance metrics of the EGPR estimation in real-time, namely: (i) the RMSE; (ii) the mean absolute percentage error (MAPE); (iii) the agreement between the model's prediction and the ground-truth (Bland-Altman plots); and (iv) the estimation delay.

Model evaluation: RMSE

Table 20 presents the RMSE (Equation 3) of the MC estimated by the EGPR model when compared to the ground truth (the three methods of indirect calorimetry), for all participants (P1 to P5), including the two trials of Participant 2 (T1 and T2), and all the activities (standing, walking at 1.5 km/h, 2.0 km/h, and 3.0 km/h, and sitting). Additionally, the last row presents the RMSE average and standard deviation for each activity.

From the results presented in Table 20, it was possible to conclude that the RMSE was higher when the MC estimated by the regression model was compared to the per-breath respirometry calculation, due to the noisy nature of this method. Additionally, due to the similarity between the steady-state and fast-estimated respirometry calculations, the RMSEs between the regression model estimation and these calculations were very similar across the different activities.

Table 20: RMSE between the MC estimated by the EGPR model and the MC estimated by each respirometry method (in W/kg), for each activity (standing, walking at 1.5 km/h, 2km/h, and 3 km/h, and sitting), and for each participant (P1-P5). The average RMSE for each activity is presented in the last row

	Per-Breath Respirometry					Steady-State Respirometry					Fast-estimated Respirometry				
															
P1	0.68	0.74	1.07	0.84	0.62	0.32	0.56	0.88	0.58	0.06	0.30	0.60	0.88	0.55	0.07
P2 (T1)	0.28	1.08	-	1.08	-	0.08	1.08	-	1.05	-	0.08	1.20	-	1.10	-
P2 (T2)	0.30	-	0.33	0.38	-	0.22	-	0.21	0.22	-	0.22	-	0.20	0.05	-
P3	0.33	1.35	1.81	1.67	0.20	0.33	1.35	1.81	1.64	0.25	0.34	1.32	1.86	1.64	0.24
P4	0.37	0.51	0.60	0.59	0.26	0.14	0.40	0.49	0.52	0.23	0.14	0.42	0.50	0.49	0.24
P5	0.40	0.38	0.31	-	-	0.27	0.31	0.27	-	-	0.28	0.34	0.26	-	-
RMSE	0.39 ±	0.81 ±	0.82 ±	0.91 ±	0.36 ±	0.23 ±	0.74 ±	0.73 ±	0.80 ±	0.18 ±	0.23 ±	0.78 ±	0.74 ±	0.77 ±	0.18 ±
	0.15	0.40	0.64	0.50	0.23	0.10	0.45	0.66	0.55	0.10	0.10	0.45	0.68	0.61	0.10

Regarding the results of the comparison to the per-breath respirometry, the average overall RMSE, for every participant and activity was 0.66 W/kg. Furthermore, the average coefficient of determination (R^2) was 0.75. Compared to the metrics obtained during the offline testing, the RMSE increased by 31.8% and the R^2 by 12%. This could be explained by the fact that the model was dealing with new data, from different subjects and different sensors. These results were analogous to another literature study that achieved a 40.4% increase in the estimation error during real-time validation, in comparison to the offline results (15).

Model evaluation: MAPE

Figure 26 presents five sets of box plots (one for each task studied) depicting the MAPEs between the MC estimated by the regression model and the estimation computed by indirect calorimetry using the per-breath respirometry method, for each participant (P1 to P5). The MAPE was calculated using Equation 4.

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{y(i) - \hat{y}(i)}{y(i)} \right| \quad (4)$$

Each box plot displays five metrics of the MAPEs for a particular activity and participant: the median MAPE (the horizontal line roughly at the center of the box), the first and third quartiles (represented by the bottom and top limits of the box, respectively) and the minimum and maximum MAPEs, excluding the outliers (represented by the limits of the whiskers, i.e. the dotted lines outside the boxes that show the range of MAPEs outside the middle 50% of data). The outliers are represented by the points outside the box plots.

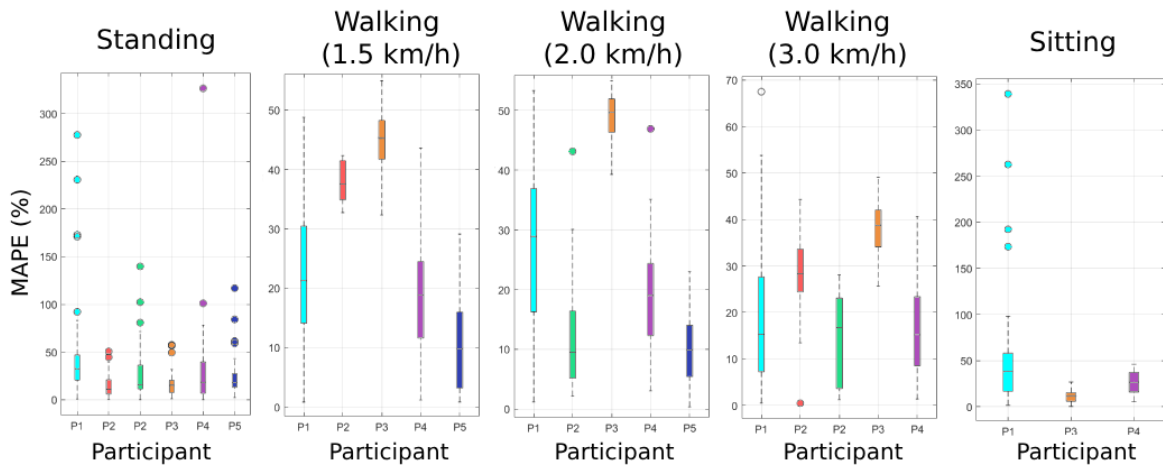


Figure 26: Boxplots presenting the MAPEs between the MC estimated by the EGPR model and the per-breath respirometry method. Each boxplot presents the results of the five participants (P1-P5) for a single activity.

From Figure 26, it was observed that the MAPE was minimal during the static activities, i.e., the standing and sitting activities. The MAPEs for the walking activities were similar, despite them being performed at different speeds, however, walking at 3 km/h was the walking task with the lower average MAPE, for all participants except Participant 2 (second trial). This could be explained by the fact that this speed was an intermediate value between the speeds performed during Ingraham’s protocol, and, therefore, the regression model was much more used to data from similar speeds.

Additionally, it was noticed that the MAPEs during the standing activity had many more outlier points. This could be explained by undesirably spontaneous movements performed by the participants during this activity that could have influenced the MC estimations. This could also explain the existence of various outliers in the MAPEs of Participant 1 during the sitting activity.

Regarding the literature studies analyzed in Table 7, only Slade et al. (15) validated the model in real-time, in a protocol with 24 participants who performed four types of activities: (i) walking; (ii) running; (iii) climbing stairs; and (iv) cycling. The overall average MAPE achieved by Slade et al. was 23%. This value is in the range of the MAPEs obtained in this work for Participants 1, 2 (second trial), 4, and 5, while Participants 2 (first trial) and 3 had an average MAPE higher than 23%.

Model evaluation: Bland-Altman plots

Figure 27 presents the Bland-Altman plot comparing the MC estimations of the EGPR model and the per-breath respirometry, for every participant. Each Bland-Altman plot shown depicts the agreement between the regression model estimation and the ground truth (the target).

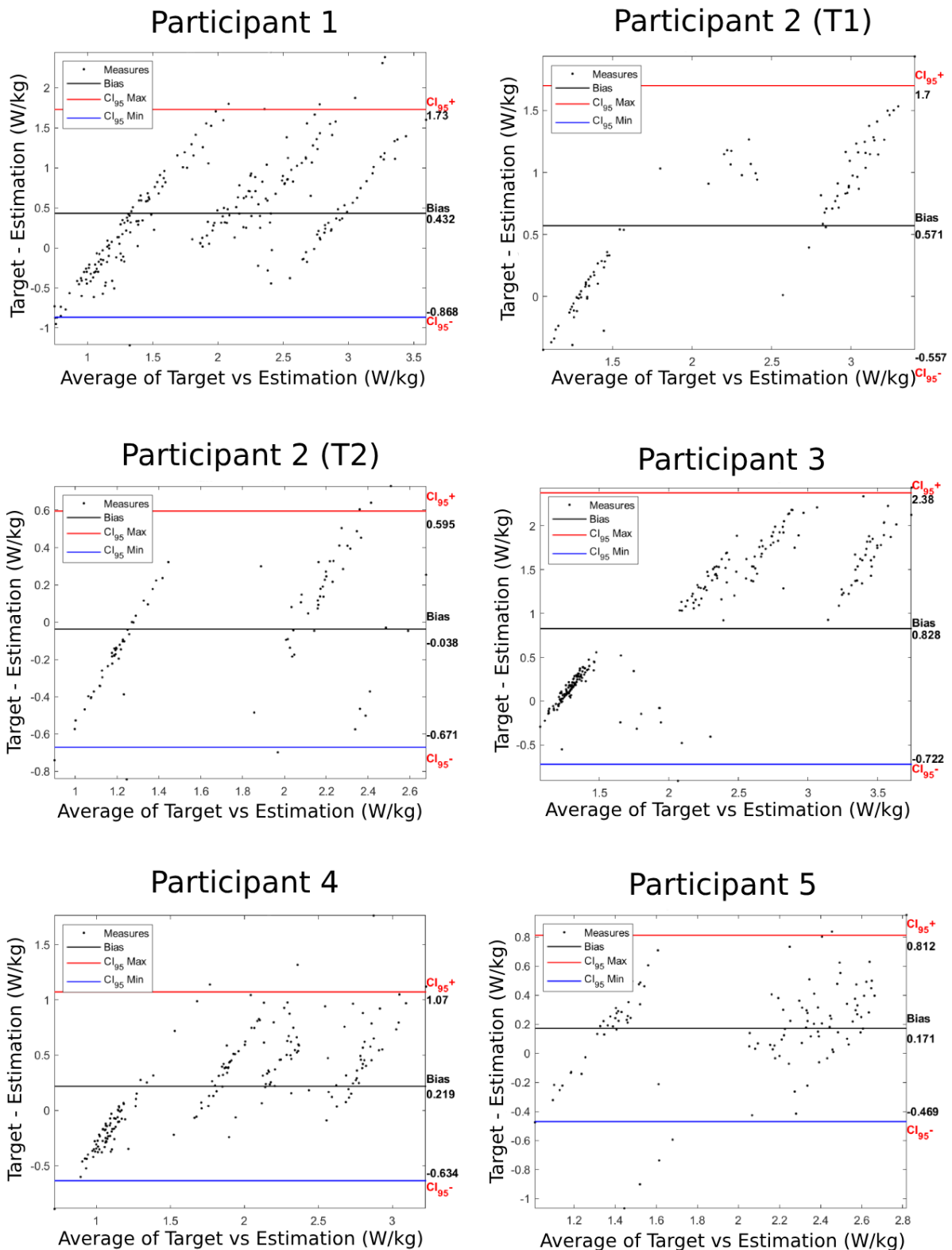


Figure 27: Bland-Altman plots depicting the agreement between the MC estimated by the EGPR model and the per-breath respirometry for each participant, where the black line represents the mean difference and the red and blue lines represent the 95% limits of agreement.

The plots of Figure 27 show the relationship between the difference between the two signals (y-axis) and the average of the two estimation methods (x-axis), for each participant. This allows the identification of estimation biases, given by the mean difference between the two signals (black line). Additionally, the upper and lower limits of agreement are also represented in the plot (red and blue lines, respectively), which are computed by the mean difference plus/minus 1.96 times its standard deviation.

Regarding the results presented by the Bland-Altman plots, they showed that the model’s prediction and the ground truth had a higher agreement for Participants 2 (second trial), 4, and 5, perceived by the smaller bias between the two signals for these participants (-0.038 W/kg, 0.219 W/kg, and 0.171 W/kg, respectively), and the smaller limits of agreement (1.27 W/kg, 1.70 W/kg, and 1.28 W/kg, respectively). With the exception of the second trial of Participant 2, whose bias was negative but close to zero, all the biases were positive, demonstrating the general underestimation made by the regression model.

Model evaluation: Estimation delay

Table 21 presents the mean delay of the EGPR model estimation and the RMSE of the time agreement between the regression model and the respirometer, for each participant (P1 to P5) and each trial of Participant 2 (T1 and T2). The delay was given by the difference between the time of each estimation made by the EGPR and the time of the previous estimation made by the respirometer (the previous breath). The time agreement RMSE was computed by comparing the estimation delay to an ideal model’s time response (no delay).

Table 21: Average delay between the estimation made by the regression model and the respirometer, and RMSE of the time agreement between the two methods, for each participant (P1-P5)

	P1	P2 (T1)	P2 (T2)	P3	P4	P5
Average estimation delay (s)	2.48 ± 2.30	1.61 ± 1.28	1.91 ± 1.88	1.64 ± 1.60	2.02 ± 1.86	1.77 ± 1.68
Time agreement RMSE (s)	3.28	2.05	2.68	2.29	2.74	2.45

The results presented in Table 21 show that the regression model estimated the MC, on average, 1.61 to 2.48 seconds later than the respirometer, and the RMSE of the time agreement was between 2.05 and 3.28 seconds. Since the average breath frequency of an adult is between 12 and 18 breaths per minute, this means that the interval between breaths is between 5 and 3.3 seconds. Therefore, the average time delay was still below the time it takes to estimate a new MC by indirect calorimetry.

Additionally, the developed model was able to estimate the MC every 10 seconds, without any delays from the SmartOs system. This is a significant upgrade when compared to a respirometer device, which takes 3 minutes of estimations before reaching a steady-state value. Therefore, the time taken by the regression model to estimate a new steady-state MC was 18 times lower than a respirometer device.

4.4.5 Conclusions

In this chapter, a regression model was developed to estimate the MC from only four IMUs, wearable and non-intrusive sensors, practical for industrial applications. The model was capable of achieving low estimation errors in real-time, similar to the metrics achieved by the best-performing models in the literature. The model was successfully integrated into the SmartOs system, with low computational cost and without affecting the system's performance. Additionally, this work allowed the reduction of the steady-state MC estimation time from 3 minutes to 10 seconds. In conclusion, the regression model was considered fit to be integrated into a HITL controller (Chapter 6).

5 Torque tracking control

The proposed HITL control presented in Chapter 3.3 is based on the premise that optimizing an exoskeleton torque profile improves the human-robot interaction and reduces the users' physical exertion. So, before implementing this HITL strategy into SmartOs' architecture, a torque tracking control was developed to manipulate the active actuator based on a reference torque profile. This chapter describes the development of this control strategy and its integration into the knee exoskeleton of SmartOs system.

Firstly, the algorithm for generating a torque profile is explained. A natural cubic spline interpolator was used for this purpose since it enabled the representation of any desired torque profile shape through the combination of multiple sinusoidal curves. The spline generates a torque profile, which is used as the reference torque trajectory for the torque tracking control.

Then, the torque control is presented, which starts with the generation of a desired torque profile and is composed of two hierarchic stages - the mid- and low-level controllers - that continuously compute the knee joint's torque. The mid-level controller was based on the continuous estimation of the gait phase based on the gait speed, and posterior calculation of the torque for that gait point. The low-level controller was a PID torque controller, a loop mechanism with the objective of minimizing the difference between the desired/reference and the real actuator's torque.

Lastly, the validation of the controller is presented. Initially, bench tests were performed to identify the best PID controller parameters (i.e., the proportional, integral, and differential gains), and the ideal knee torque profile for gait assistance. Then, a human experimental validation enabled the assessment of the HITL strategy's effectiveness.

5.1 Introduction

Most of the torque tracking controllers in the literature are developed for pneumatic actuators or tethered exoskeletons (8, 49, 51), not for LLEs with electric motors. Pneumatic or cable-driven actuators' controllers are implemented to follow the torque trajectory of healthy humans. However, this strategy is not the most adequate for electric actuators as DC motors, due to their different mechanical system.

Some studies in the literature that developed torque tracking controllers for ankle exoskeletons with electric motors created time-adaptive control algorithms that generate torque profiles with adjustable torque timings and magnitudes, based on the device's movement (93) (94). The torque profile implemented by these studies had to be different from the natural human torque profile, in order for the electric motor to perform the desired movement.

Therefore, when developing a torque tracking controller for an electric actuator, the torque profile that will be used as a reference trajectory must be manually adapted for the required application. Figure 28 presents the differences between the ankle torque profile of a tethered actuator (8) and a DC motor (93), as well as the natural torque profile for the ankle joint from a healthy person (95).

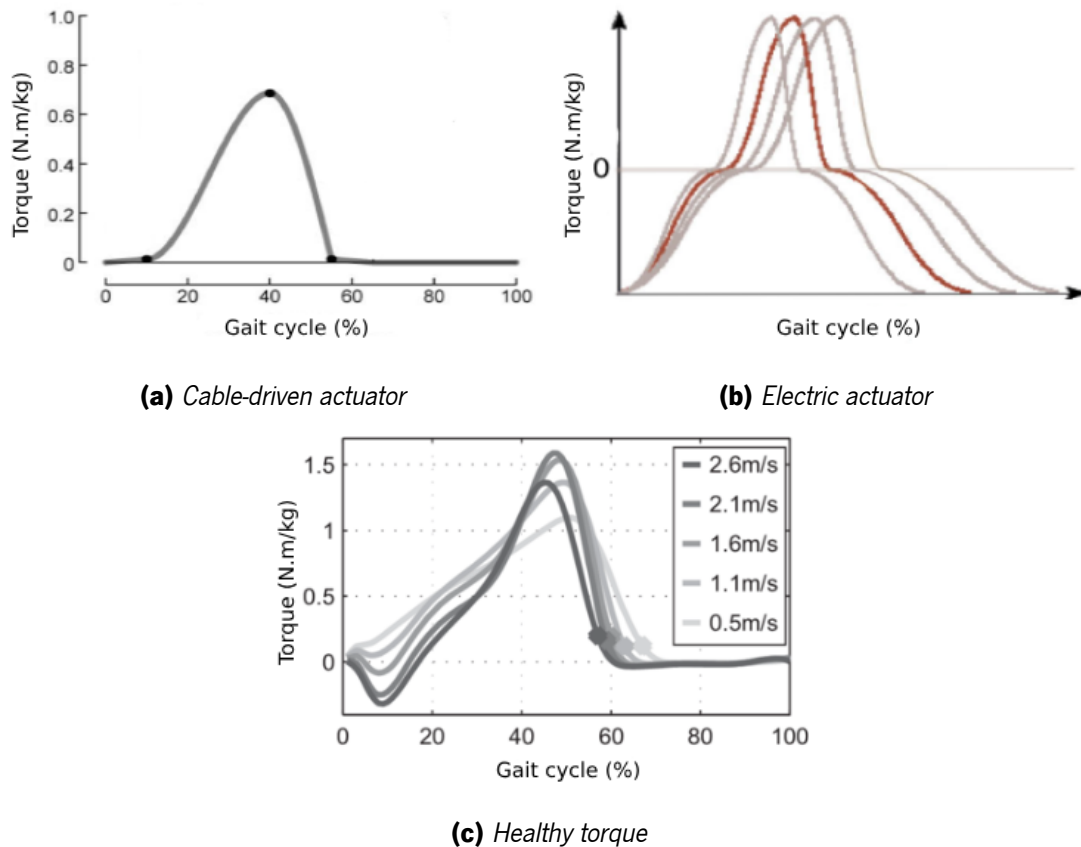


Figure 28: Ankle torque profiles used on torque tracking controllers for a tethered exoskeleton (8) and an exoskeleton with electric actuators (93), and the natural ankle torque profile (95)

Figure 28 shows that the torque profile used by the tethered device (Figure 28a) tries to replicate the natural ankle torque (Figure 28c) by simplifying the curve to one single positive curve. However, that is not the case for the torque profile used by the electric exoskeleton (Figure 28b), which was composed of two torque curves, one positive and one negative, with similar torque magnitudes. This profile was set in order to ensure the movement of the actuator in both dorsiflexion and plantarflexion directions, so the device performs a similar ankle trajectory to the natural ankle trajectory of a human. Therefore, in order to perform the desired ankle movement, the torque profile of the electric actuator ended up differing from the natural human torque profile. This was the approach followed during the development of the torque tracking control developed in this chapter.

5.2 Methods

5.2.1 Torque profile generation

The knee torque profile is the variation of the torque magnitude in time, during one gait cycle. Figure 29 presents the natural human torque, of the knee joint, when walking at speeds from 0.5 m/s (1.8 km/h) to 2.6 m/s (9.4 km/h) (95). Figure 29 shows that the natural knee torque is mainly composed of two positive curves, that generate the joint's flexion movement, and one negative curve, responsible for the extension movement.

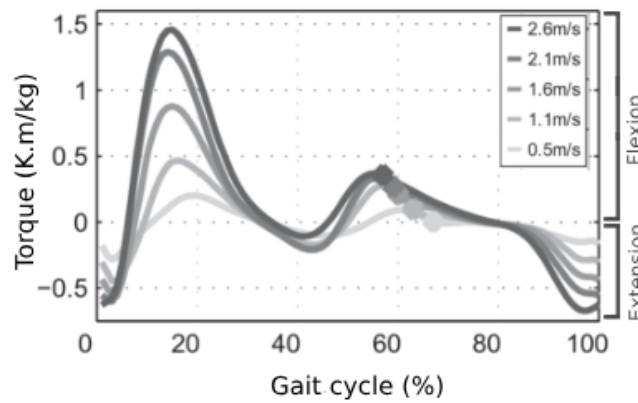


Figure 29: Human knee torque profile when walking at different speeds (95).

The knee torque profile can be simplified by a combination of multiple positive and negative sinusoidal curves, that represent the flexion and extension movements of the knee joint, respectively. Therefore, it is possible to use mathematical algorithms to generate this curve, such as a natural cubic interpolator.

The development of the torque profile described in this chapter started, therefore, with the creation of an algorithm capable of generating the knee torque profile in C by natural cubic interpolation, and its implementation in the LLOS controller, following the guidelines in the literature (96). The Algorithm 1 is the pseudocode that computes the torque profile's parameters, where n is the number of points that define the spline, x is an array with those points' gait cycle percentages, and a is the torque magnitude in those points ($a_i = T(x_i)$).

After the torque profile's parameters are obtained, it is possible to determine any point in the torque profile by knowing its gait cycle percentage. Equation 5 shows how the torque magnitude - $T(X)$ - for the point X is determined, knowing that X value ranges between x_i and $x_{(i+1)}$.

$$T(X) = a_i + b_i(X - x_i) + c_i(X - x_i)^2 + d_i(X - x_i)^3 \quad (5)$$

Algorithm 1 Natural cubic spline algorithm

SET $n, x = [x_0, x_1, \dots, x_n], a = [a_0, a_1, \dots, a_n]$

SET $l_0 = 1, u_0 = 0, z_0 = 0$

for $i = 1$ to $n - 1$ **do**

SET $h_i = x_{(i+1)} - x_i$

SET $\alpha_i = 3 \frac{(a_{(i+1)} a_i)}{h_i} - 3 \frac{(a_i a_{[i-1]})}{h_{[i-1]}}$

SET $l_i = 2(x_{(i+1)} - x_{(i-1)}) - h_{(i-1)} u_{(i-1)}$

SET $u_i = \frac{h_i}{l_i}$

SET $z_i = \frac{\alpha_i - h_{(i-1)} z_{(i-1)}}{l_i}$

end for

SET $l_n = 1$

for $j = n - 1$ to 0 **do**

SET $c_j = z_j - u_j c_{(j+1)}$

SET $b_j = \frac{(a_{(j+1)} - a_j)}{h_j - h_j (c_{(j+1)} + 2c_j) / 3}$

SET $d_j = \frac{c_{(j+1)} - c_j}{3h_j}$

end for

Figure 30 presents one possible knee torque profile, as an example, with two flexion torque peaks (8 N.m and 18 N.m) and one extension torque peak (-16 N.m). This profile replicates the general shape of the natural knee torque profile (Figure 29) for slow walking speeds (1.8 km/h).

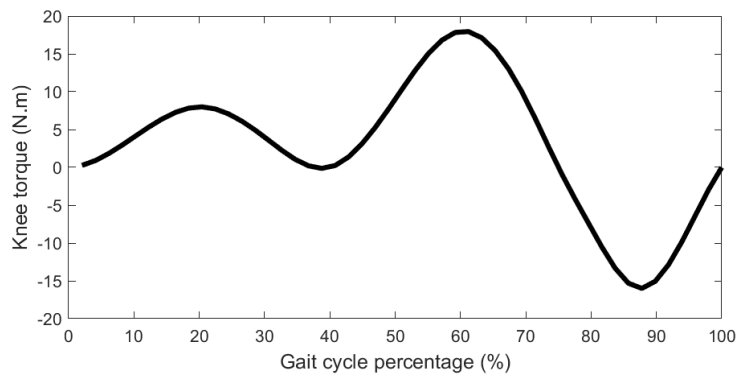


Figure 30: Curve shape of a torque profile with two flexion movements, with torque magnitude peaks at 20% and 60% of the gait cycle, and one extension movement with a torque magnitude peak at 87.5% of the gait cycle.

5.2.2 Torque controller implementation

Figure 31 presents the overall diagram of the torque tracking controller integrated into SmartOs' architecture. The code was fully developed in the LLOS board, however, the SmartOs' CCU is needed to start the device, and the mobile APP to configure its assistance.

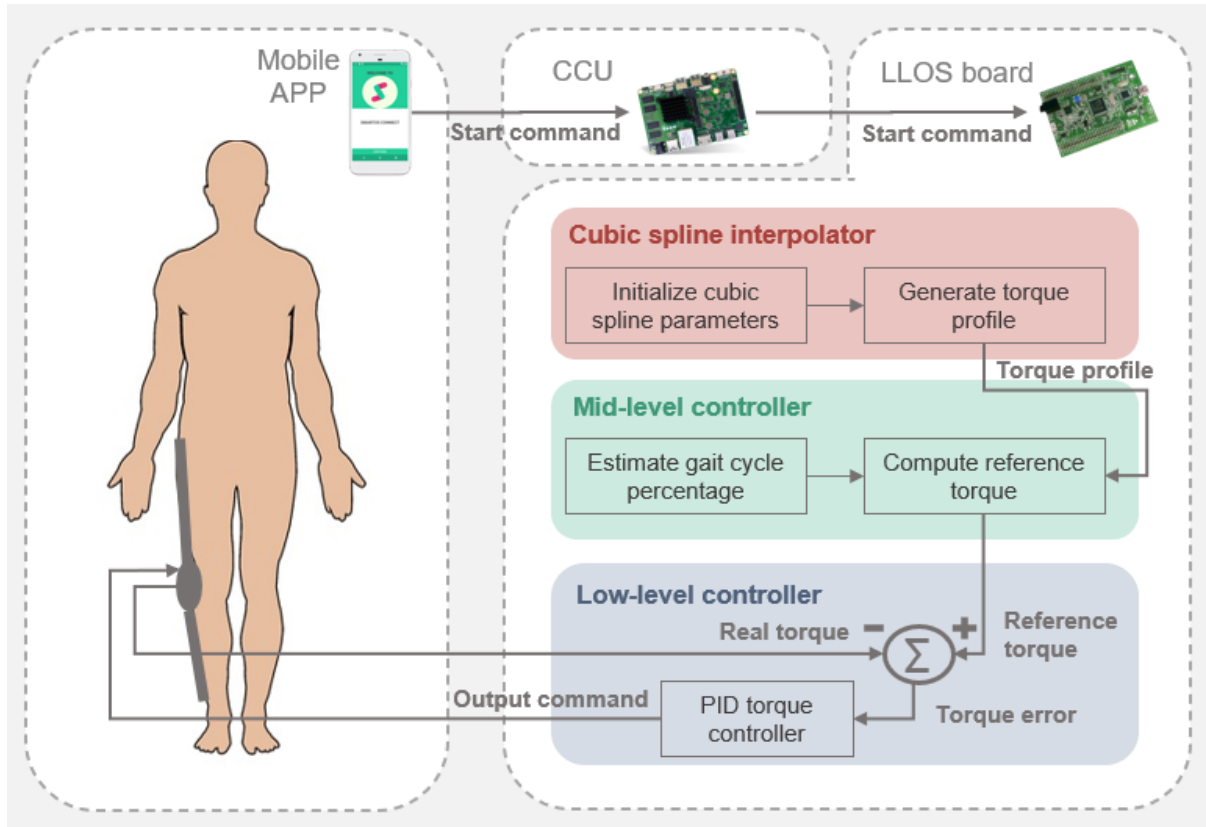


Figure 31: Diagram of the torque controller integrated into SmartOs' architecture.

The controller starts by generating the torque profile that will be replicated by the SmartOs' knee actuator. Then, every 10 milliseconds, in the mid-level stage of the controller, the gait cycle percentage is estimated and the reference torque is obtained, i.e., the instant torque of the torque profile for the estimated gait cycle percentage (Equation 5). The low-level controller is executed every millisecond and it runs a PID controller that repeatedly measures the torque error (i.e., the difference between the reference and the real motor torque) and feeds the active actuator with the torque command required to reduce this error. The following paragraphs will better explain each block of the torque controller.

Cubic spline interpolator

The code of the cubic spline interpolator is composed of two simple steps that are executed when starting the SmartOs' device before the knee actuation is activated. Firstly, the cubic spline is initialized

and the arrays that define the torque profile (x and a) are established. Then, the torque profile is generated by following Algorithm 1.

Mid-level control

In the mid-level controller, an algorithm to compute the reference torque for the PID low-level controller was implemented. This algorithm is executed every 10 milliseconds, something made possible by a timer. The mid-level controller can be divided into two phases: the detection and action phases.

In the detection phase, the algorithm estimates the gait cycle phase, given by a percentual point (0% refers to the start of a gait cycle, while 100% refers to its end). This estimation is done by using an equation used to obtain the duration of one gait cycle in milliseconds (gcd) - Equation 6 - based on the walking velocity in km/h (v). The algorithm then increments the gait cycle percentage every 10 ms by following Equation 7, where gcp_i is the new gait cycle percentage and gcp_{i-1} is the percentage previous to that one. The $fmod()$ function is used to calculate the floating-point remainder of $gcp_i/100$, setting the maximum ceiling of the gait cycle percentage to 100%.

$$gcd = (-34,62v + 107.31) \times \frac{49}{1000} \quad (6)$$

$$gcp_i = fmod(gcp_{i-1} + \frac{100 \times 10 \text{ ms}}{gcd}, 100) \quad (7)$$

The action phase is based on a force profile control strategy, where the torque is computed for each gait cycle percentage and transmitted to the low-level controller. This step uses Equation 5 to calculate the torque magnitude of the torque profile for each gait cycle point.

Low-level control

A PID controller was implemented at the low-level stage of the LLOS controller, which was executed every millisecond due to the implementation of a timer. Algorithm 2 presents the PID controller's algorithm.

The PID controller's objective is the minimization of the difference between the reference torque and the real torque, i.e., the controller error, where the real torque is the knee actuator's torque measured by a Hall sensor installed in the DC motor. The algorithm starts by setting the proportional, integral, and derivative gains (kp , ki , and kd , respectively), and initializing the error integral (et). Every millisecond, the torque error is computed, its time integral is updated, and the output torque is obtained. The error's integral was saturated at 40 N.m.s. This output torque is then sent to the knee actuator motor.

Algorithm 2 PID controller algorithm

```
SET  $kp, ki, kd$   
SET  $error_i = ref\_torque - real\_torque$   
UPDATE  $et = et + error_i$   
if  $et > 40$  then  
    SET  $et = 40$   
else if  $et < -40$  then  
    SET  $et = -40$   
end if  
SET  $output\_torque = error_i \times kp + et \times ki + (error_i - error_{i-1}) \times kd$ 
```

Furthermore, the PID output was limited to values between -2500 and 2500 . Additionally, to guarantee the users' safety, the knee angle was limited to values between 15° and 80° . This was ensured by assessing the PID output and the knee angle after the execution of Algorithm 2.

SmartOs mobile APP

The SmartOs' mobile APP was altered to allow the activation of the developed torque controller. For this purpose, an additional option was added to the assistance settings page of the APP called 'Therapy Torque'. Figure 32 depicts the assistance settings page when the torque controller is selected for the right knee exoskeleton and walking speeds of 1.5 km/h.

5.3 Validation

5.3.1 Bench tests

The validation started with the tuning of the PID controller, i.e., with the adjustment of the proportional, integral, and differential gains to ensure a quick and adequate response from the low-level controller's stage. This was achieved by following the Ziegler–Nichols method (97).

Additionally, the knee torque profile was also continuously adjusted to identify the ideal torque assistance, in a manual process. During this process, it was ensured a correct and comfortable gait pattern by the SmartOs' user.

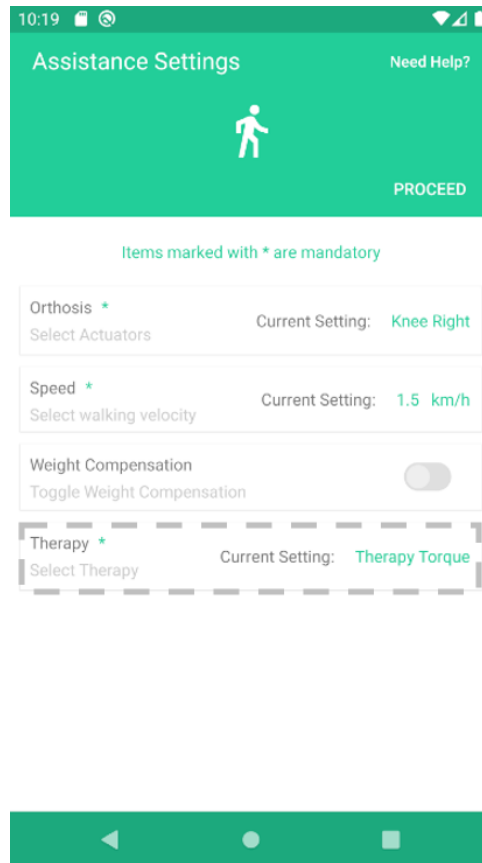


Figure 32: Interface option added to the SmartOs' APP that enabled the start of torque control.

5.3.2 Human experiments

Following the bench tests used to assess the operability of the torque controller and tune the controller's parameters, an experimental validation was conducted to assess the human biomechanics and physiological state when assisted by the knee module of SmartOs' controlled by the developed torque strategy. Additionally, the device behavior was also evaluated, namely regarding its motor's torque conformity to the reference torque and the PID command output.

In this phase, one volunteer was equipped with the SmartOs' knee exoskeleton on the right leg. Furthermore, the participant wore the four InertiaLab's IMUs required by the regression model, previously described in Chapter 4, to estimate the MC of the participant in real-time.

Participants

One healthy volunteer participated in this validation phase after giving her informed consent. The participant had no history of locomotor or balance impairment nor did she suffer any musculoskeletal injury six months prior to this experiment. The participant was a 23-year-old female, with a body mass

of 65 kg and a height of 1.62 m. The participant had no previous experience in wearing the SmartOs' device. This protocol was conducted under the ethical procedures of the Ethics Committee in Life and Health Sciences (CEICVS 006/2020), following the Helsinki Declaration and the Oviedo Convention.

Experimental protocol

The participant was first equipped with four InertiaLab's IMUs in the right ankle, right wrist, left waist, and chest. Then, the knee module of SmartOs' was tightly secured to the participant's right leg, with three straps - one at the upper leg, and two at the lower leg - and one belt at the waist. The IMUs were then connected to the WMSS board through USB cables and the knee actuator was connected to the LLOS board through a CAN Bus cable. Afterward, both the WMSS and LLOS boards were connected to the SmartOs' CCU, and the system was plugged into the SmartOs' battery. Figure 33 presents an overview of the equipment worn by the participant during the protocol.

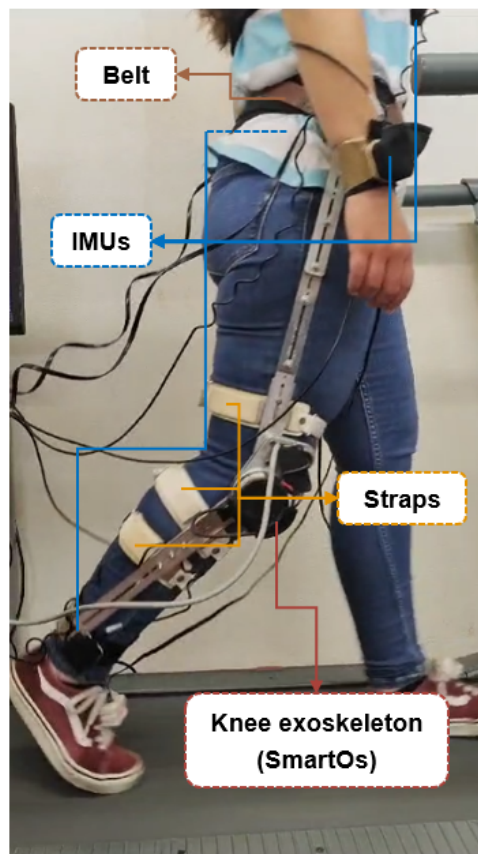


Figure 33: *Equipment worn by the participant during the torque controller's validation.*

Before the participant was subjected to the torque controller, she went through a familiarization period to get comfortable and experienced in wearing the SmartOs' knee device. Firstly, the participant walked with the device in zero-torque mode, a control strategy that makes the active actuator mimic the functioning

of a passive actuator (i.e., it is the person who controls the device). Then, the participant walked with the device in position tracking control, a reference-tracking controller where the reference signal is the knee trajectory during a gait cycle. This familiarization process lasted until the participant felt comfortable wearing and walking with the device.

Then, the participant was finally prepared to test the developed torque controller. Firstly, the participant stood still for 5 seconds to calibrate the IMUs. After the calibration, the participant walked on a treadmill at 1.5 km/h with the SmartOs in torque tracking control. This procedure lasted for a total of 5 minutes.

Data collection and analysis

The protocol was concluded in one day, and performed at the Biomedical Robotic Devices Laboratory (BirdLab) facility, at the University of Minho. The data acquired during this protocol comprised of (i) the 3D acceleration of the right ankle, right wrist, left waist, and chest; (ii) the estimated MC; (iii) the right knee angle; (iv) the human-robot interaction torque; (v) the real actuator's torque; and (vi) the PID output commands. The data collected was saved during the acquisition into text files, in real-time, at a frequency of 100 Hz.

The data was then processed and analyzed in MATLAB (2022b, The Mathworks, Natick, MA, U.S.A). The processing procedure consisted of organizing the collected data in a single table. Then, there were created graphs to analyze the controller's performance.

5.4 Results and discussion

5.4.1 Bench tests

Regarding the tuning of the proportional, integral, and differential gains (k_p , k_i , and k_d , respectively), the optimal found values are presented in Table 22. The PID tuning was conducted by following the Ziegler-Nichols method (97). Firstly, the integral and differential gains were set to zero and the proportional gain increased until a stable output was achieved. Then, the integral and differential gains were found based on the optimal proportional gain.

Table 22: Best PID gains values

Gain	k_p	k_i	k_d
Value	135	1.5	1.5

The proportional gain ended up being significantly higher than the other gains, resulting in larger command outputs for smaller torque errors. Regarding the integral and differential gains, the rather small values in comparison to the k_p were sufficient to diminish overshoot and oscillations and fix any offset.

In regards to the optimal torque profile for gait assistance, various shapes, with various torque magnitudes at the flexion and extension peaks, were generated and tested until a good pattern was established, i.e., a profile that ensured a correct and comfortable gait pattern. Figure 34 presents the various torque profiles that were tested and their order (from the first to the sixth profile), depicting, therefore, the torque profile's evolution during this procedure. The peak torque magnitudes for each profile differed, and various profiles were tested for each shape presented, by changing the x and a vectors of Algorithm 1. To simplify, only the torque magnitudes for flexion and extension peaks of 15 N.m and -15 N.m, respectively, are presented.

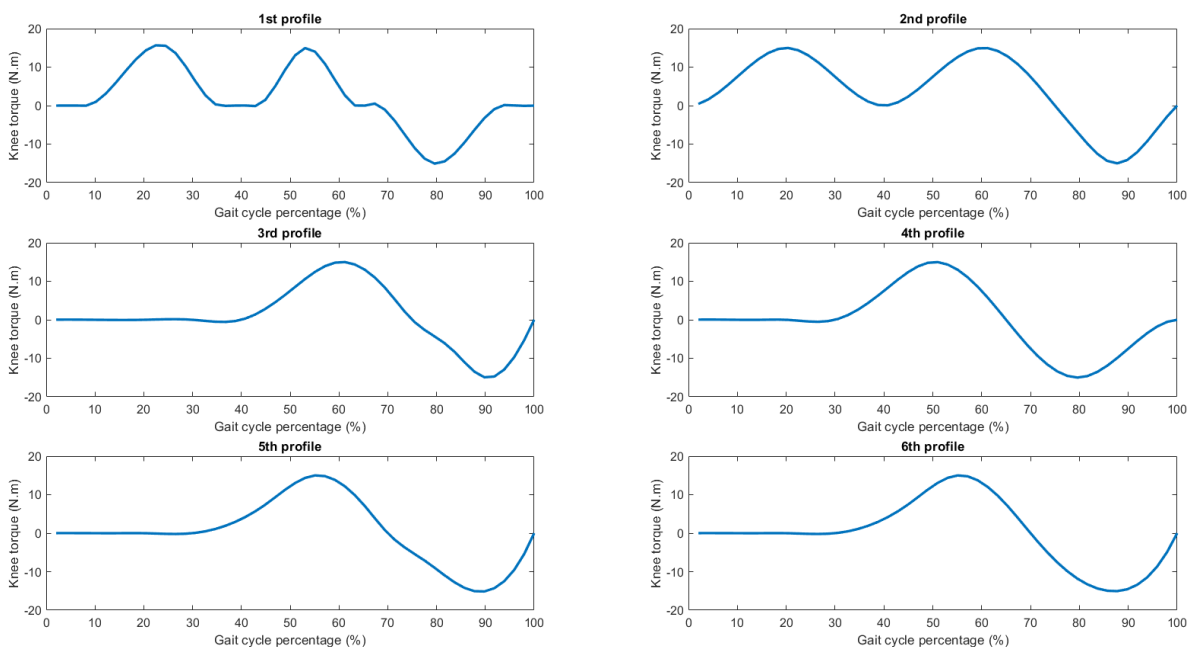


Figure 34: Tested torque profiles, for flexion and extension peak torques of 15 N.m and -15 N.m, respectively.

Figure 35 presents the torque profile that achieved the best results, both in terms of the capability of the PID controller to achieve the desired pattern and on the generated gait's quality, evaluated by one user's perceived comfort level. The profile is composed of a positive curve (flexion movement) with a peak of 20 N.m of magnitude at 55% of the gait cycle, and a negative curve (extension movement) with a peak at 86% of the gait cycle and a magnitude of -20 N.m.

Regarding the best torque profile, it was observed that a curve with only one positive and a negative peak felt more natural to the user and could be better replicated by the PID low-level controller. Despite not perfectly reproducing the natural knee torque pattern of a healthy person, the profile presented in Figure

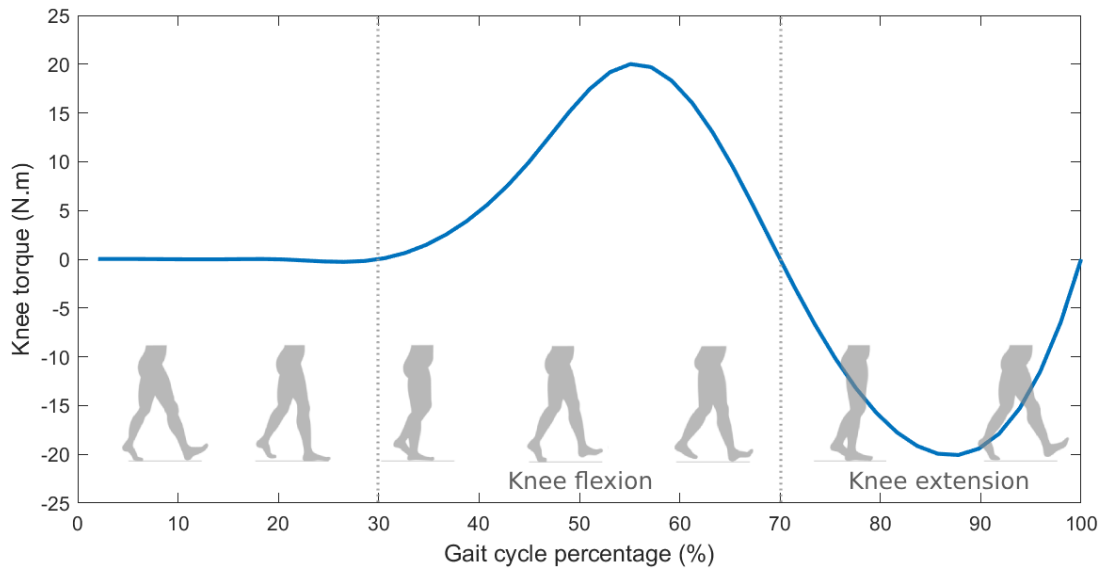


Figure 35: *Best performing knee torque profile.*

35 allowed for a full flexion movement and a subsequent extension movement that restored the leg to the original position. Furthermore, the timings of each movement enabled the user to walk comfortably at any gait speed between 0.5 km/h and 1.5 km/h (i.e. the SmartOs' speed range). This work followed the approach of other studies that developed torque controllers for DC motors, that found a torque reference that drove the device to follow a desired trajectory (93, 94).

5.4.2 Human experiments

The following section presents the results achieved during the experimental protocol performed to validate the torque controller. The results' analysis focused on the PID's performance and the physiological and biomechanical signals of the user.

PID's performance

Figure 36 shows the performance of the PID low-level controller. The figure depicts the reference torque profile (blue, dotted), the real motor torque (blue, solid), and the PID command (orange, solid) during the first 30 seconds of data.

From this analysis, it was possible to conclude that the SmartOs' system was capable of following the torque trajectory without delay. Additionally, the participant was able to walk at 1.5 km/h in a comfortable gait pattern, fully synchronized with the knee exoskeleton. Therefore, the main requirements of the torque tracking controller were verified.

These results indicate that the motor torque was incapable of achieving the 20 N.m torque magnitude.

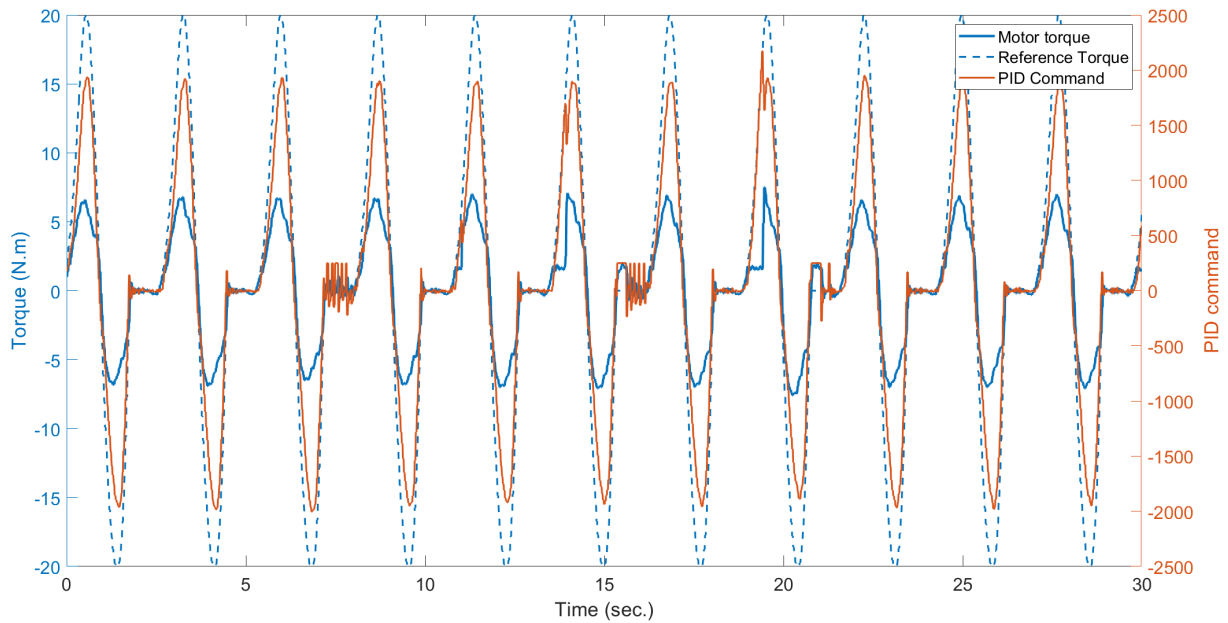


Figure 36: Actuator and reference torques during 30 seconds of torque control (colored blue, solid and dotted, respectively) and the PID command output (colored orange).

Despite the device's torque curve shape being mostly identical to the torque pattern, the maximum torque magnitudes were roughly around 7 N.m and -7 N.m for the flexion and extension movements. This was due to limitations regarding the SmartOs' DC motor and was impossible to correct. Despite this limitation, the motor torque was still sufficient to assist the device's user during a normal gait.

Figure 36 also shows that around some zero-torque zones (i.e., the stance phase) the PID command suffered some oscillations. This was due to a safety feature that conditions the PID command to increase when the knee angle goes below 15 degrees, to force the motor to move in the opposite direction and increase the knee angle. Therefore, this was not prejudicial to the system and even proved that the safety feature was operational.

Furthermore, it was observed that some of the reference torque's positive curves' shapes were not perfectly replicated by the motor, namely in the 6th and 8th gait cycles presented in Figure 36. In these two cycles, the motor torque 'blanked' for some milliseconds, at the start of the extension movement, causing the device to interrupt the movement momentarily. This could be explained by the mechanical limitations of the system caused by the actuator's latency. Still, the controller was able to respond fast to match the intended torque reference. These anomalies were quick and almost imperceptible, and the participant was still perfectly capable of walking at 1.5 km/h by increasing the interaction with the device to move to the desired position when these motor failures occurred.

Physiological and biomechanical signals

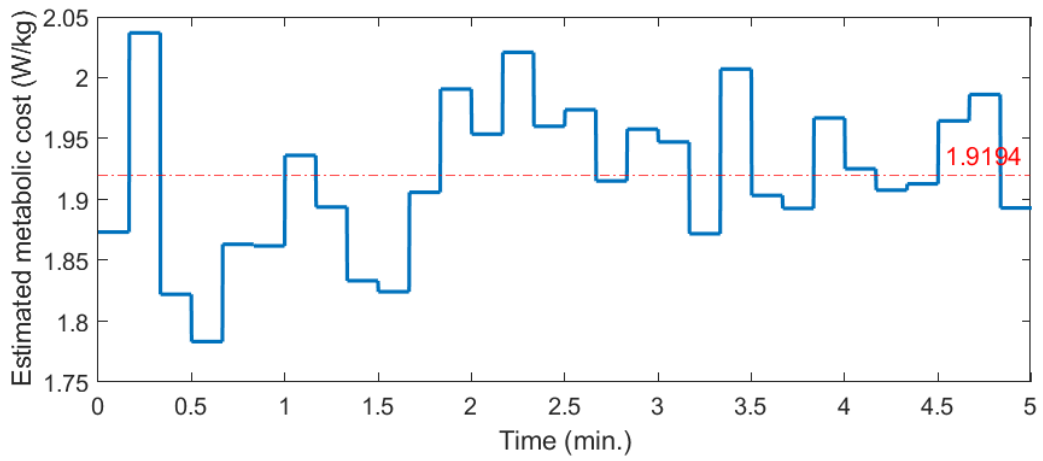
Figure 37 presents the physiological and biomechanical analysis of the user when walking with the device in torque control mode, at 1.5 km/h. Figure 37a depicts the variation of the estimated MC during the 5-minute protocol, as well as a horizontal red line representing the MC average. Figure 37b presents the human-robot interaction torque, with two red lines referent to the maximum and minimum interaction torque during the gait. Figure 37c shows the knee angle during the experiment.

Figure 38 presents the interaction torque and knee angle measured during a smaller period of 30 seconds (between minutes 3.5 and 4 of the 5-minute walking trial), corresponding to 12 gait cycles. Figure 38a depicts the participant's interaction torque, while Figure 38b the knee angle.

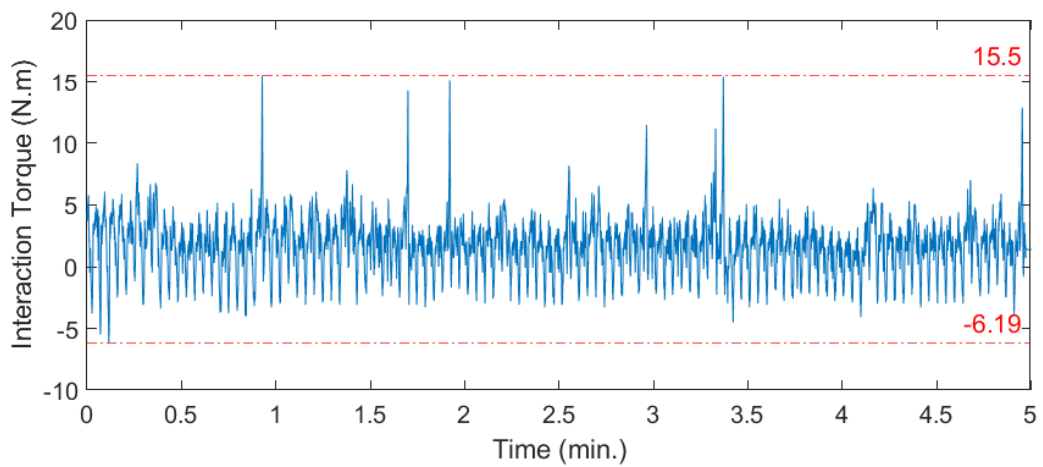
Regarding the person's MC, Figure 37a shows that the participant's MC varied significantly at the start of the experiment (first 2 minutes), and was more stable towards the final 3 minutes. This initial instability could be explained by the fact that the participant was still getting used to wearing the device with the torque controller and performed some abrupt movements when trying to gain balance since the movement was still quite unstable. Nonetheless, even after the 2-minute mark, the person's MC was still fluctuating, however, this fluctuation was not abnormal to the MC estimating regression model and was generally below 0.05 W/kg, therefore, it was considered insignificant. The average MC was 1.92 W/kg, which is a high value for walking speeds of 1.5 km/h when compared to walking without the exoskeleton (as seen in Figure 25). The high MC could be explained by the high torque of the system, which resulted in fast leg movements, increasing the ankle acceleration and, thus, the MC.

In regards to the participant's interaction torque (Figures 37b and 38a), it was observed that, in general, this metric was rather small - between 7 N.m and -4 N.m. However substantial outlier positive values were measured at some points during the gait, setting the maximum interaction torque to be 15.5 N.m. These abnormalities could be explained by the motor torque's glitches identified in Figure 36. As mentioned before, when these motor failures occurred, the participant was forced to increase the interaction torque to push the device in the correct trajectory. However, ignoring these rare mishaps, the participant's torque was significantly lower than the natural torque of a 65 kg person when free-walking, without an exoskeleton (52 N.m during flexion and -13 N.m during extension, as presented in Table 10).

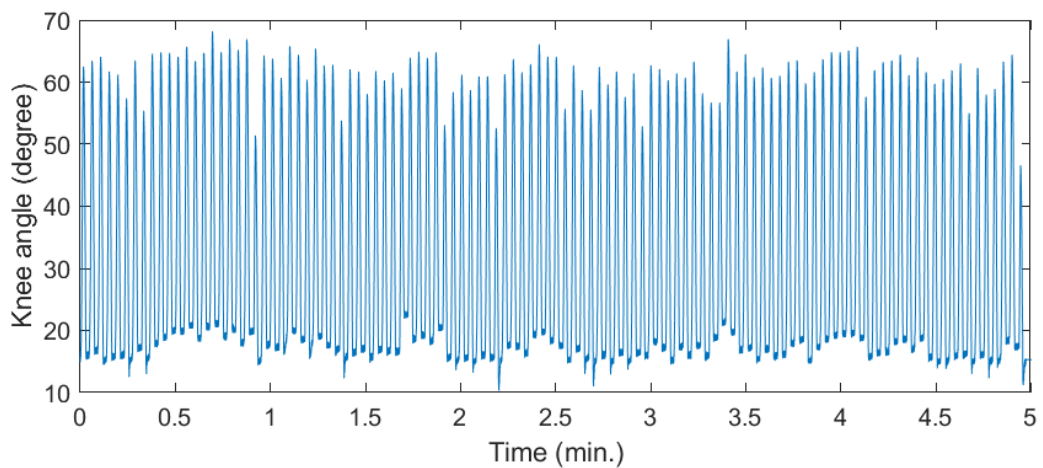
Regarding the knee angle, presented in Figures 37c and 38b, it was verified that the torque controller enabled a continuous and regular knee's trajectory. Additionally, it was observed that the knee angle was successfully maintained above 15 degrees, which was a safety and operational requirement of the DC motor. Despite the natural knee's ROM during walking activities being 60 degrees (Table 9), it was concluded that the participant could still comfortably walk at 1.5 km/h with a ROM of 50 degrees.



(a) Estimated MC (dotted red line presents the variable's average)

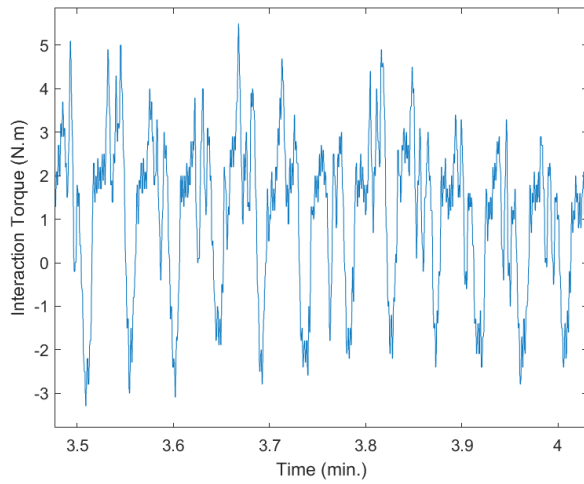


(b) Measured interaction torque (dotted red lines present the variable's maximum (top) and minimum (bottom))

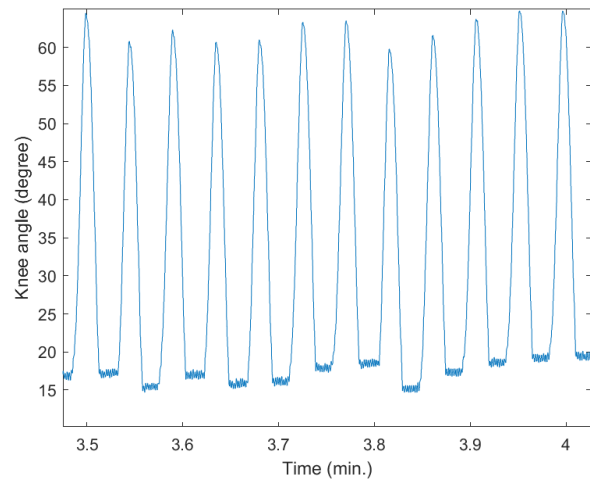


(c) Measured knee angle

Figure 37: Physiological and biomechanical variables measured during the human experimental protocol.



(a) Measured interaction torque



(b) Measured knee angle

Figure 38: Participant's interaction torque and knee angle, measured during 30 seconds (between 3.5 and 4 minutes).

5.5 Conclusions

This chapter presented the development of a torque tracking control for a knee exoskeleton, which was successfully integrated into the SmartOs' architecture. Firstly the ideal torque profile was found to guarantee a comfortable gait by the exoskeleton's user. The controller was capable of following the desired torque profile, by estimating the gait phase, computing the reference torque, and driving the actuator to minimize the difference between its torque and the reference. The user was able to accurately perform the desired knee movement during a gait at 1.5 km/h. In conclusion, this controller was considered adequate to be integrated into the HITL controller.

6 Human-in-the-loop control

This chapter describes the development of a HITL control capable of adapting the reference torque control to minimize the effort of a person wearing the SmartOs system. The effort will be measured by the user's MC and the user's interaction torque with the device. The work presented here followed the development of a regression model used to estimate the MC in real-time, presented in Chapter 4, and the implementation of a torque control strategy in SmartOs' architecture, presented in Chapter 5. The regression model was capable of obtaining an estimation of the MC based on the 3D acceleration measured by four IMUs on the right ankle, left waist, right wrist, and chest. The torque control was developed to make the knee DC motor follow a desired torque profile.

The HITL controller adapted two parameters from the knee torque profile: the torque magnitudes of the flexion and extension peaks. This work started with the establishment of the knee torque profile shape, which was based on a natural cubic spline with two peaks on predefined fixed points of the gait cycle and fixed durations.

Then, the optimizer used to find the knee torque profile that leads to the minimal value of the objective function is described. The optimizer used in this work was a CMA-ES optimizer, an evolutionary algorithm that mimics the process of natural selection, and adapts the reference torque over various generations to find the fittest solution. The objective function optimized by the algorithm was a weighted sum of the users' MC, the torque profile's integral, and the user's interaction torque with the device. The CMA-ES was integrated into the SmartOs' architecture, more specifically into the LLOS board, and combined with the torque controller previously presented.

Finally, the validation of the HITL controller is also presented here. Firstly, a bench test was performed to analyze possible time constraints between the optimizer and the mid- and low-level controllers and optimize the CMA-ES objective function's weights. Then, the effectiveness of the controller was studied, i.e., its capability of minimizing the MC of a person, the torque profile's integral, and the interaction torque, in real-time.

6.1 Introduction

Several literature studies have proved that wearing an LLE can result in a reduction of the MC of their users when the right assistance strategies are employed, such as HITL controllers (8). One possible HITL strategy is the adaptation of the exoskeleton joints' torque profiles, in real-time, to minimize the MC of a person (52).

This automatic and individualized optimization approach has been successful in various published studies. However, most of the studies that developed a HITL controller used a respirometer device to estimate the MC in real-time by indirect calorimetry. This approach is not practical to implement in real-world applications due to the discomfort that comes with wearing these masks, the cost of the materials, and the time restrictions associated with this slower MC estimation method (15, 16).

The most common alternative to indirect calorimetry is the use of machine or deep learning models to estimate the MC based on signals acquired by one or more wearable sensors (98), however, to the authors' best knowledge, no study so far has integrated this approach into a HITL control strategy. Additionally, the HITL controllers developed for portable exoskeletons are scarce in the literature, and most of the optimization algorithms take too long to compute the optimal control parameters (8, 13, 19).

The HITL controller presented here aims to tackle the limitations found in the existing strategies present in the literature. Firstly, the controller was developed for the SmartOs device, which is an LLE with an 8-hour autonomy. The controller minimizes an MC estimated by a regression model fed by data from only four wearable, non-intrusive and light IMUs, replacing the need for a respirometer device. The use of the regression model developed in Chapter 4 also enabled the reduction in the time needed to estimate the MC from 3 minutes to 10 seconds, therefore, significantly decreasing the time required to reach the optimal control parameters.

Furthermore, when designing a controller that aims to minimize users' physical effort it is also important to reduce the burden placed on their joints. So, in addition to the MC, the controller should minimize the interaction torque and the reference torque profile's integral final value (i.e., the total system's torque during a gait cycle). When the reference torque's integral after one gait cycle is zero, the knee joint's final position will be the same as its initial position, meaning that the user's movement is fully assisted by the exoskeleton. The interaction torque is another metric that measures the user's effort when wearing the exoskeleton. When the exoskeleton is capable of achieving the desired trajectory without the users' help, the person's exertion will be reduced, and the user-exoskeleton interaction torque will be minimal.

6.1.1 Covariance Matrix Adaptation – Evolutionary Strategy

In this section, the optimizer implemented in the HITL controller is described. The CMA-ES is a derivative-free randomized search algorithm that can be used for black-box scenarios where a certain objective function needs to be optimized by adapting the covariance matrix of a multivariate normal distribution (99, 100). CMA-ES is an evolutionary strategy as it stochastically samples a fixed number of

individuals (also known as candidates) in every generation, based on the results of the individuals of the previous generation, and evaluates their fitness (the value of the objective function), to have superior individuals every generation (12, 55).

The multivariate normal distribution that is updated every generation (g) is represented by a mean and covariance matrix that depends on the objective function values measured by that generation's individuals ($N(m, C)$, where m is the mean, and C the covariance matrix). Every generation, a new set of individuals is chosen when sampling this distribution, by following Equation 8 (12, 99), where $x_k^{(g+1)}$ is the k^{th} individual from generation $g+1$, $m^{(g)}$ is the search distribution's mean value for generation g , $\sigma^{(g)}$ is the standard deviation/step-size at generation g , $C^{(g)}$ is the covariance matrix for generation g , and λ is the number of individuals per generation.

$$x_k^{(g+1)} \sim m^{(g)} + \sigma^{(g)} N(0, C^{(g)}) \text{ for } k = 1, \dots, \lambda \quad (8)$$

For the optimization algorithm to have information on all previous generations, an evolution path (p_c) that saves the relation between consecutive steps (generations) is implemented in CMA-ES. Additionally, the scale of the normal distribution – its step-size – is also increased or decreased every generation. The step-size is also controlled by an additional evolution path (p_σ) (99).

Overall, after the fitness value of every candidate in a generation g is obtained, the distribution is updated by updating the five state variables of the optimizer: (i) the mean of the search distribution, $m^{(g+1)}$, obtained using Equation 9; (ii) the step-size evolution path, $p_\sigma^{(g+1)}$, by using Equation 10; (iii) the evolution path of the covariance matrix, $p_c^{(g+1)}$, which is calculated by using Equation 11; (iv) the covariance matrix, $C^{(g+1)}$, by using Equation 12; and (v) the step-size itself $\sigma^{(g+1)}$ by using Equation 13 (99).

$$m^{(g+1)} = \sum_{i=1}^{\mu} \omega_i x_{i:\lambda}^{(g+1)}, \quad \sum_{i=1}^{\mu} \omega_i = 1, \quad \omega_i > 0 \quad (9)$$

$$p_\sigma^{(g+1)} = (1 - c_\sigma) p_\sigma^{(g)} + \sqrt{c_\sigma(2 - c_\sigma) \mu_{eff}} \frac{m^{(g+1)} - m^{(g)}}{\sigma^{(g)}} \quad (10)$$

$$p_c^{(g+1)} = (1 - c_c) p_c^{(g)} + \sqrt{c_c(2 - c_c) \mu_{eff}} \frac{m^{(g+1)} - m^{(g)}}{\sigma^{(g)}} \quad (11)$$

$$C^{(g+1)} = (1 - c_{cov}) C^{(g)} + \frac{c_{cov}}{\mu_{cov}} p_c^{(g+1)} p_c^{(g+1)T} + c_{cov} \left(1 - \frac{1}{\mu_{cov}} \right) \times \sum_{i=1}^{\mu} \omega_i \left(\frac{x_{i:\lambda}^{(g+1)} - m^{(g)}}{\sigma^{(g)}} \right) \left(\frac{x_{i:\lambda}^{(g+1)} - m^{(g)}}{\sigma^{(g)}} \right)^T \quad (12)$$

$$\sigma^{(g+1)} = \sigma^{(g)} \exp \left(\frac{c_\sigma}{d_\sigma} \left(\frac{E \| p_\sigma^{(g+1)} \|}{E \| N(0, I) \|} - 1 \right) \right) \quad (13)$$

Various variables are used from Equations 9 to 13, namely the recombination weights (ω_i), the learning rate for the cumulation step size control (c_σ), the variance effective selection mass (μ_{eff}), the learning rate for the cumulation for the rank-one update of the covariance matrix (c_c), the learning rate for the matrix update (c_{cov}), the weighting parameter between rank-one and rank- μ update (μ_{cov}), the damping parameter for step-size update (d_σ), the expectation value (E), and the normal distribution with zero mean and unity covariance matrix ($N(0, I)$) (99).

The first step when optimizing a certain function with CMA-ES is the initialization of the optimizer variables. Hansen (99) published a default strategy to set these variables, which used Equations 14 to 22, where n denotes the search space dimension:

$$\lambda = 4 + 3 \lfloor 3 \ln n \rfloor \quad (14)$$

$$\mu = \lfloor \lambda / 2 \rfloor \quad (15)$$

$$\omega_i = \frac{\ln(\mu + 1) - \ln i}{\sum_{j=1}^{\mu} (\ln(\mu + 1) - \ln j)} \quad \text{for } i = 1, \dots, \mu \quad (16)$$

$$\mu_{eff} = \frac{1}{\sum_{i=1}^{\mu} \omega_i^2} \quad (17)$$

$$c_\sigma = \frac{\mu_{eff} + 2}{n + \mu_{eff} + 3} \quad (18)$$

$$d_\sigma = 1 + 2 \max \left(0, \sqrt{\frac{\mu_{eff} - 1}{n + 1}} \right) + c_\sigma \quad (19)$$

$$c_c = \frac{4}{n + 4} \quad (20)$$

$$\mu_{cov} = \mu_{eff} \quad (21)$$

$$c_{cov} = \frac{1}{\mu_{cov}} \frac{2}{(n + \sqrt{2})^2} + \left(1 - \frac{1}{\mu_{cov}} \right) \min \left(1, \frac{2\mu_{eff} - 1}{(n + 2)^2 + \mu_{eff}} \right) \quad (22)$$

The algorithm ends when a termination criterion is met. Various termination conditions can be set, such as limiting the number of generations or limiting the fitness function values that can be measured. These conditions are normally application-dependent and are chosen according to the user's needs.

6.2 Methods

6.2.1 Knee torque profile

As previously explained, only the peak torque magnitudes of the knee torque profile are adapted in real-time by the CMA-ES algorithm to minimize the exoskeleton user's effort, meaning that the shape of the profile stays consistent. This was done following the results achieved by several studies that proved that when optimizing a torque profile to minimize the MC of several distinct individuals, the peak torque magnitudes of the optimized profile observed more fluctuations than the time parameters across the different participants (8, 12, 14, 51).

Figure 39 presents the knee torque profile that will be optimized by the HITL control. This profile was based on the results previously obtained in Chapter 5.4.1. The points adapted by the CMA-ES algorithm, in real-time, are represented by a red dot. The points of the profile that intersect the x-axis (the zero-torque points from 0-30% and at 70%) were fixed. Figure 39 shows that the positive and negative peaks are executed at 55% and 86% of the gait cycle, respectively, and the flexion and extension torques have a duration of 40% and 30% of the gait cycle, respectively.

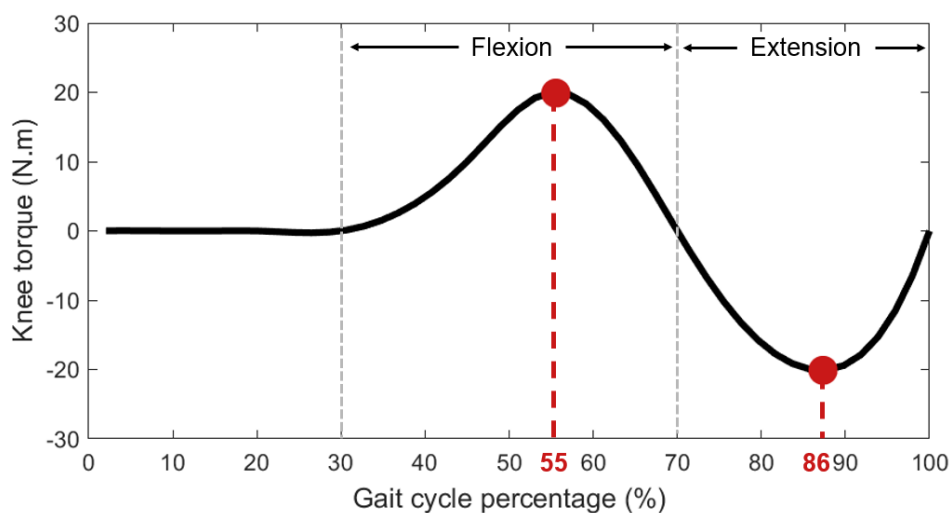


Figure 39: Curve shape of the knee torque profile that was optimized by the HITL controller.

6.2.2 CMA-ES optimizer

For the torque profile to be adjusted in real-time and for the controller to learn the effect of each change on the user's exertion, a CMA-ES optimizer was then implemented in the LLOS board. The code was based on a publicly available repository (101), adapted and condensed to be integrated into the LLOS controller and fit the existing code, using the Keil μ Vision 5.0 IDE. Algorithm 3 presents the pseudocode of a generic CMA-ES optimizer.

Algorithm 3 CMA-ES optimizer algorithm

```
INITIALIZE optimizer's parameters
while !termination() do
    COMPUTE  $\lambda$  new candidate solutions
    SET  $i = 0$ 
    repeat
        GET  $(i + 1)$ th candidate
        GET  $(i + 1)$ th fitness function value
        INCREMENT  $i$ 
    until  $i = \lambda$ 
    UPDATE function value history
    UPDATE  $m, p_\sigma, p_c, C, \sigma$ 
end while
```

Table 23 presents the optimizer's initial parameters. These parameters were chosen following the recommended strategy identified in the literature (99). Furthermore, Table 23 presents the initial torque peaks studied by the optimizer: 17.5 and -17.5 N.m, for the peak flexion magnitude and the peak extension magnitude, respectively ($start_1$ to $start_2$). Additionally, it also presents the range of possible torque magnitudes for each peak. The parameters max_1 and max_2 are the maximum possible magnitudes for the flexion and the extension peak torques, respectively. The min_1 and min_2 are the minimum magnitudes for the flexion and the extension peak torques, respectively. The initial step-size (σ) was chosen based on the literature (55).

Table 23: Initial parameters' values for the CMA-ES optimizer. Based on: (99)

n	2	μ	3
λ	6	c_σ	0.51
d_σ	1.51	c_c	0.7
c_{cov}	0.12	μ_{cov}	2.24
σ_1	1.4	max_1	20 N.m
σ_2	1.4	max_2	-15 N.m
σ	1.4	min_1	15 N.m
C	Identity matrix	min_2	-20 N.m
ω_1	0.59	$start_1$	17.5 N.m
ω_2	0.29	$start_2$	-17.5 N.m
ω_3	0.12	μ_{eff}	2.24

Seven termination conditions were set on the algorithm. The optimization process stopped when:

1. The fitness function value reached a smaller value than the **StopFitness** parameter.
2. The fitness function value difference between two consecutive steps was smaller than the **ToIFun** parameter.
3. The fitness function value of the best values in two consecutive generations was smaller than the **StopToIFunHist** parameter.
4. The step-size, in the x-space, was smaller than the **ToIX** parameter.
5. The step-size, in the x-space, increased by more than the **ToIUpXFactor** parameter.
6. The maximum number of generations (**StopMaxIter**) is reached.
7. The maximum number of function evaluations (**StopMaxFunEvals**) is reached.

The termination parameters were all set to the code default values, apart from the maximum number of iterations which was set to 20, and the maximum number of function evaluations, which was set to $900(n + 3)(n + 3)$. Table 24 presents the values used for the termination conditions' parameters.

Table 24: Values of the parameters used for the termination conditions of the CMA-ES optimizer.

StopFitness	0	TolUpXFactor	1e3
StopTolFun	1e-12	StopMaxIter	20
StopTolFunHist	1e-13	StopMaxFunEvals	22500
TolX	1e-11		

6.2.3 HITL control implementation

Figure 40 presents the overall diagram of the HITL control integrated into SmartOs' system. This figure presents the general data flow between the different elements of SmartOs' architecture: (i) the InertiaLab's sensors, (ii) the WMSS board, (iii) the CCU, (iv) the LLOS board, (v) the mobile APP, and finally (vi) the powered knee actuator.

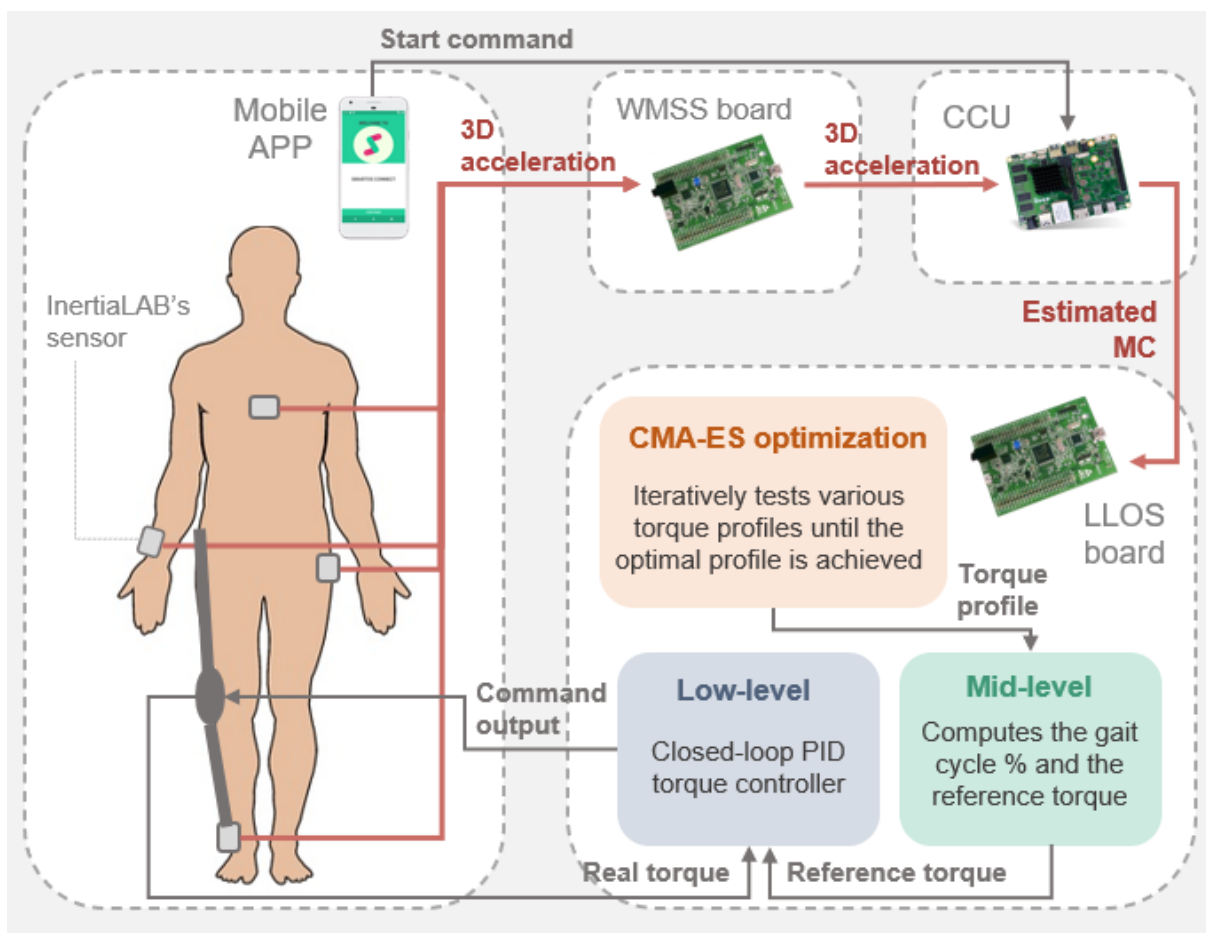


Figure 40: Diagram depicting the different blocks of the HITL control strategy developed in SmartOs' architecture.

The code developed for the HITL controller unified the codes previously described in Chapters 4.2.4

(regression model developed for MC estimation) and 5.2.2 (general torque controller for a fixed torque profile) and combined them with a CMA-ES optimizer. The regression model algorithm was consistent with the fluxogram presented in Figure 12 with the exception of one crucial addition: the transmission of the estimated MC to the LLOS board where the HITL controller is executed after each estimation is completed (every 10 seconds).

As seen in Figure 40, the code developed in the LLOS board was divided into three blocks: (i) the CMA-ES optimizer in orange; (ii) the mid-level controller in green; and (iii) the low-level controller in blue. This strategy is a version of the torque controller presented in Chapter 5.2.2, where the torque profile is continuously adjusted by the CMA-ES algorithm. Therefore, the mid- and low-level stages of this controller are identical to the torque controller previously integrated into SmartOs' architecture. The following section further explains the CMA-ES optimization stage of the controller.

CMA-ES optimizer

Figure 41 presents the code fluxogram of the CMA-ES optimizer, where IT and TI represent the interaction torque and reference torque integral, respectively. This algorithm was based on the Algorithm 3 presented in Chapter 6.2.2.

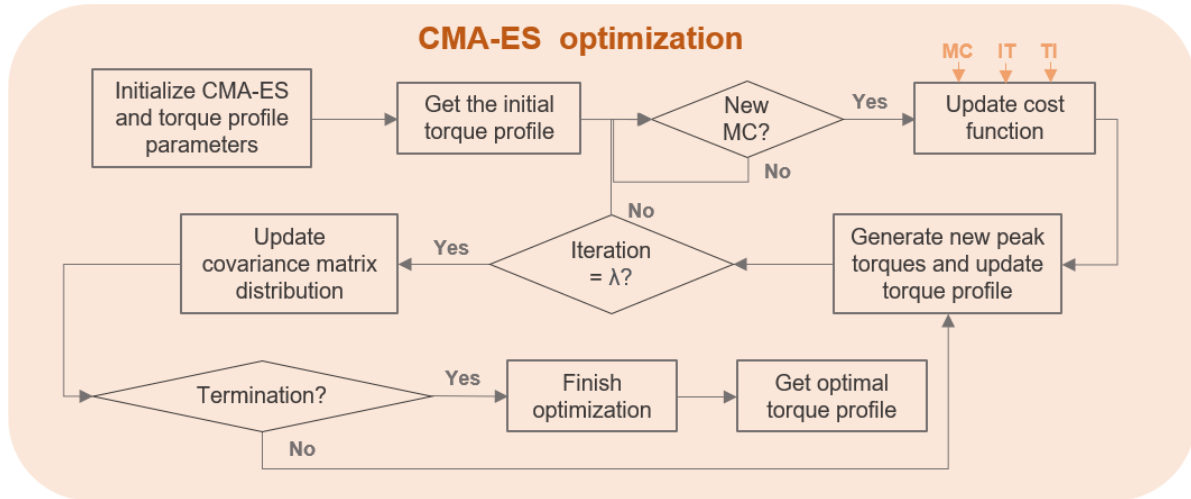


Figure 41: Code fluxogram of the CMA-ES optimizer developed in the LLOS board.

However, some adjustments had to be made in order to combine the optimization process with the MC estimation happening on the system's CCU. The purpose of this code block was to generate multiple combinations of peak torque magnitudes and study each torque profile's effect on the person's exertion. The person's exertion was evaluated through an objective function computed through the weighted sum of three parameters: (i) the estimated MC; (ii) the generated torque profile's integral (TI); and (iii) the

user's interaction torque (IT). The objective function (OF) value at each CMA-ES iteration is computed using Equation 23, where ω_1^{OF} , ω_2^{OF} , and ω_3^{OF} are the objective function weights for each parameter.

$$OF = \omega_1^{OF} \times MC + \omega_2^{OF} \times TI + \omega_3^{OF} \times IT \quad (23)$$

The code starts with the initialization of the CMA-ES algorithm and the parameters of the natural cubic spline that was developed to compute the knee torque profile in Chapter 5.2.1. This initialization is done prior to the start of the system actuation and is followed by the acquirement of the initial torque profile (profile presented in Figure 39 with peak torque magnitudes of 17.5 N.m and -17.5 N.m), allowing for the control kickoff.

Every time a new MC estimation is received, the optimizer's cost function is updated and a new torque profile is generated and applied to the DC motor. This is repeated for λ iterations, and, when each generation is over (each set of λ repetitions) the covariance matrix distribution is updated and the optimizer parameters are updated by using Equations 9 to 13.

When a termination condition is met, the optimization is finalized and the optimal magnitudes of the peak torques are calculated. Afterward, these values can be saved and the optimal torque profile is continuously applied to assist the person. Therefore, this optimization process is individual-specific and only needs to be performed once per individual.

SmartOs mobile APP

Afterward, the mobile APP that serves as an interface with the SmartOs' system was also modified to allow the activation of the HITL control strategy, by adding a "Human-In-The-Loop" option in the assistance strategy settings of the APP. Since the HITL assistance is dependent on the MC estimated by the regression model, when the "Human-In-The-Loop" option is chosen in the APP, the "MC Estimation" button is also toggled automatically, which, according to the changes presented in the Chapter 4.2.4, also forces the communication with the InertiaLab's sensors through the WMSS board. Figure 42 presents the modification made to the APP, as well as the additional settings that are activated when the HITL control is chosen.

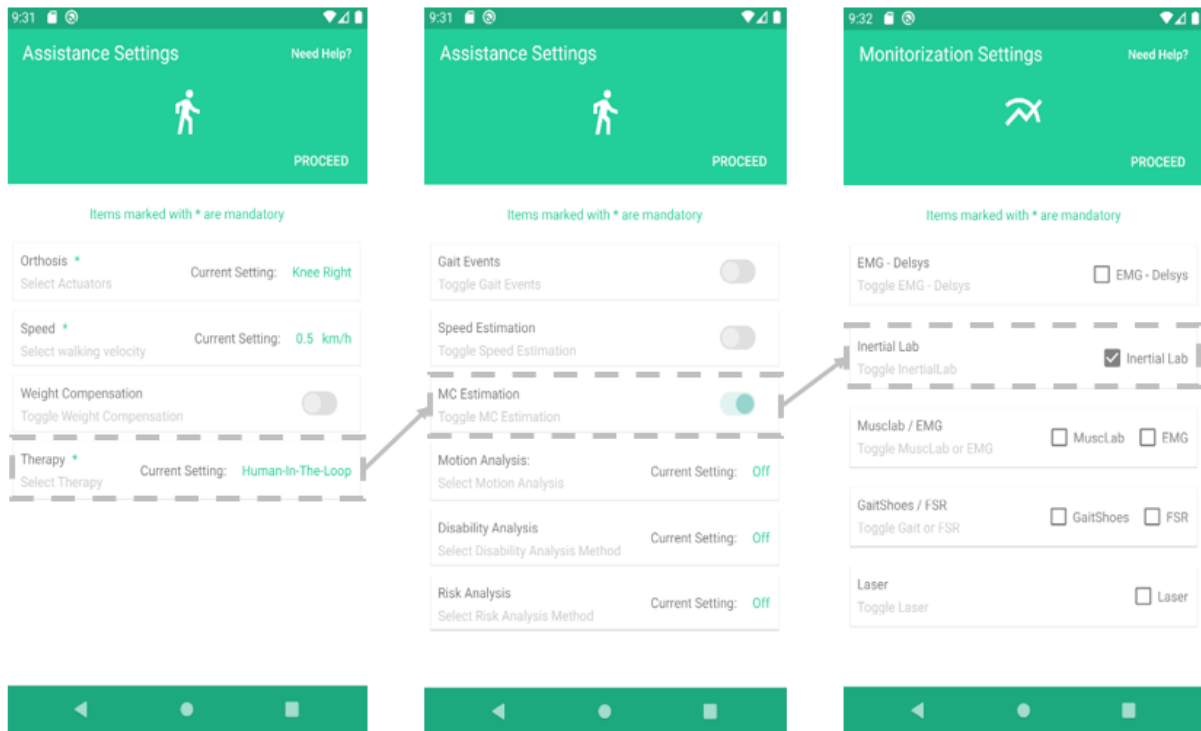


Figure 42: Interface option added to SmartOs' APP that enabled the start of HITL control and its relation to the previous adjustments.

6.3 Validation

6.3.1 Bench tests

To ensure the correct execution of the HITL controller some preliminary tests were performed and the algorithm's timings were analyzed. This validation process was done by analyzing the time taken to execute each function of the CMA-ES optimizer. To do this, an output pin of the LLOS board was activated every time each function was called and deactivated when it ended. The timings of the algorithm were then obtained by inspecting this pin in an oscilloscope. The results were organized in a time diagram to verify any time constraints in the code.

Furthermore, an additional bench test was performed to tune the CMA-ES objective function's weights, the parameters ω_1^{OF} , ω_2^{OF} , and ω_3^{OF} of Equation 23. This was done by conducting various HITL optimizations, with different sets of weights (from 0.5 to 2.0 each), and evaluating the optimal torque profile generated and the variation of the objective function and the three optimized variables over the optimization time.

6.3.2 Human experiments

After verifying the viability of the HITL controller and tuning the CMA-ES objective function to match its application needs, a human experimental validation was performed to assess the effectiveness of the HITL strategy. For this purpose, a torque profile was first optimized in real-time using the CMA-ES algorithm, for one participant. The next day, the participant walked with the optimized torque controller, with a zero-torque controller already existent in the SmartOs' architecture (that mimics the functioning of a passive actuator), and walked without wearing the knee exoskeleton. The efficacy of the optimized solution was evaluated by measuring the estimated MC and the interaction torque of the participant during these three conditions.

Participants

This experimental validation was performed by one volunteer who gave its informed consent to the protocol. This participant was healthy, had no history of locomotion or balance impairments, and did not suffer from any recent musculoskeletal injury. The participant was a 23-year-old female, with a body mass and height of 65 kg and 1.62 m, respectively, and was moderately experienced in wearing the SmartOs' knee module. This protocol was conducted under the ethical procedures of the Ethics Committee in Life and Health Sciences (CEICVS 006/2020), following the Helsinki Declaration and the Oviedo Convention.

Experimental protocol

This protocol was split into two days. On the first day, the participant went through the HITL optimization process, used to assess the optimal torque profile of a torque controller, specific to that individual. On the next day, three conditions were tested and compared: (i) no-exoskeleton; (ii) zero-torque mode; and (iii) optimized torque control.

On the first day, the participant was first equipped with four InertiaLab's IMUs in the right ankle, right wrist, left waist, and chest. Afterward, the participant equipped the knee module of SmartOs, which was well secured to their right leg with three straps at the upper and lower leg and one belt at the participant's waist. Then, the SmartOs architecture's subsystems were all connected, namely the IMUs, the WMSS board, the LLOS board, the CCU, and the active-actuated knee module. The system was then plugged into the SmartOs' battery. Figure 43 depicts the participant wearing the described equipment.

Then, the HITL controller's optimization started. The participant first stood still for 5 seconds to calibrate the IMUs and then started walking at 1.5 km/h on a treadmill. This process lasted until a CMA-ES termination condition was verified. The optimizer ended up undergoing the 20 maximum generations,

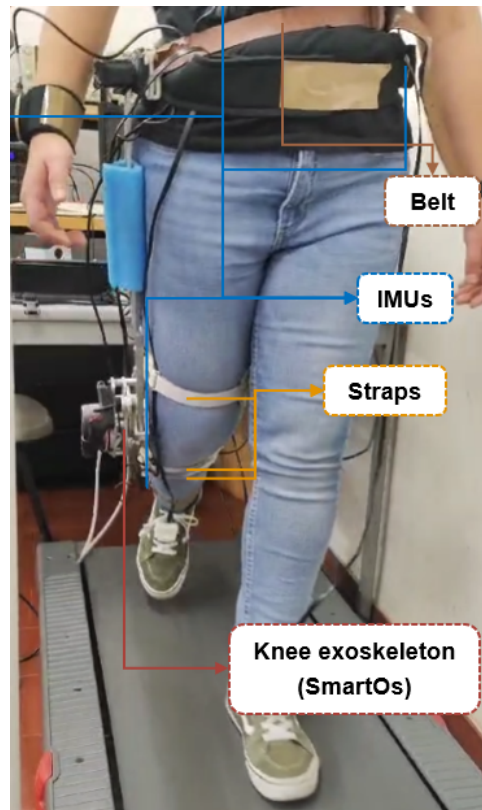


Figure 43: *Equipment worn by the participant during the HITL controller's validation.*

on a total of 120 iterations (20×6), and lasted, therefore, 20 minutes since each iteration took 10 seconds to complete.

On the second day, the participant was first equipped with the InertiaLab's sensors. Then, she walked on the treadmill for 5 minutes at 1.5 km/h, after concluding the required 5 seconds for IMUs calibration. In this first phase, the participant was not assisted by the SmartOs' knee module, thus only her MC was evaluated.

After walking without the exoskeleton, the participant was finally equipped with the SmartOs' device, and the whole system was connected and turned on as previously described. The participant walked again on the treadmill for 5 minutes, at 1.5 km/h, with the SmartOs' assistance in the zero-torque mode. This control strategy was already implemented in the architecture to make the active actuator mimic the functioning of a passive actuator (i.e., it is the person who controls the device). To conclude the protocol, the participant then walked at 1.5 km/h, for a total of 5 minutes, with the device controlled by the optimized assistance profile that was identified on the previous day. Both these experiments were preceded by the 5-second IMUs calibration.

Data collection and analysis

The experimental validation was fully performed at the Biomedical Robotic Devices Laboratory (BirdLab) facility, at the University of Minho. The data acquired during the two days was composed of: (i) the MC estimated by a regression model; (ii) the 3D acceleration of the right ankle, right wrist, left waist, and chest; (iii) the SmartOs' right knee angle; (iv) the human-robot interaction torque; (v) the SmartOs' motor torque; and (vi) the CMA-ES algorithm objective function. The data was collected and saved into text files in real-time, during the protocol, at a frequency of 10 milliseconds.

The data processing and analysis were performed in MATLAB (2022b, The Mathworks, Natick, MA, U.S.A). Regarding the first day's data, i.e. the CMA-ES optimizer data, the torque magnitudes of the torque profile's peaks were first collected and organized per iteration. Then the estimated MCs, the interaction torques cumulative sums, the integrals of the reference torques, and the objective function values were also organized per each CMA-ES iteration, and associated with the correspondent torque's magnitudes on a table. The processing of the second day's data consisted of organizing the collected data in a single table and transforming it to present the time in minutes.

6.4 Results and discussion

6.4.1 Bench tests

Figure 44 presents the time diagram of the CMA-ES optimizer's code implemented in the LLOS board. The diagram depicts how much time each CMA-ES function takes to be completely executed, and shows the timings analysis from the time a new MC estimation is received at the LLOS board (every 10 seconds), to the time that a new CMA-ES generation is started. As explained before, in Chapter 6.2.3, the algorithm only checks for termination after the λ iterations of a single generation are completed, and, if no termination condition is verified the algorithm establishes a new generation by computing new λ iterations. Therefore, the diagram depicts the worst-case scenario of the algorithm, regarding its execution time, since it shows the condition where a generation has finished and a new one is computed.

The results show that each execution of the algorithm, which is induced by a new MC, takes a maximum of 3.546 milliseconds. This is a positive result since it is a value well below the 10-second requirement, ensuring that each CMA-ES iteration is fully completed before a new MC is received. Furthermore, despite the maximum running time of the algorithm being above 1 millisecond, the frequency of the low-level controller, the CMA-ES algorithm does not interfere with this controller's stage on account of both codes being executed on different threads, therefore, running in parallel.

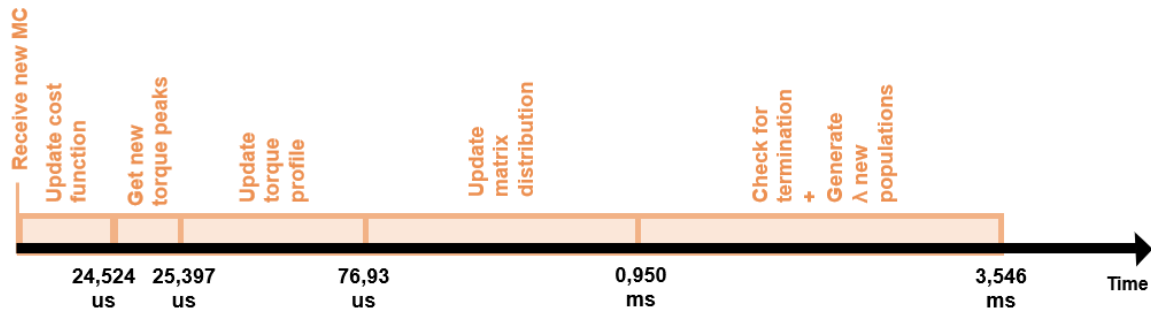


Figure 44: Time diagram depicting the time taken by the SmartOs system to execute the different HITL control algorithm's functions.

Table 25 presents the optimal CMA-ES objective function's weights. The selected weight for the estimated MC (ω_1^{OF}) and the interaction torque's cumulative sum (ω_3^{OF}) was 1 and the weight for the reference torque's integral (ω_2^{OF}) was 0.5. These values were manually obtained by performing various HITL optimizations, during treadmill walking at 1.5 km/h with the SmartOs' powered knee module and evaluating the CMA-ES behavior during the optimization.

Table 25: Best objective function weights

Weight	ω_1^{OF}	ω_2^{OF}	ω_3^{OF}
Value	1	0.5	1

The results show that the same importance was given to the estimated MC and interaction torque, regarding their impact on the CMA-ES objective function. The minimization of these two physiological parameters in an equal manner was imperative for good CMA-ES performance since both signals have an analogous impact on the user's exertion.

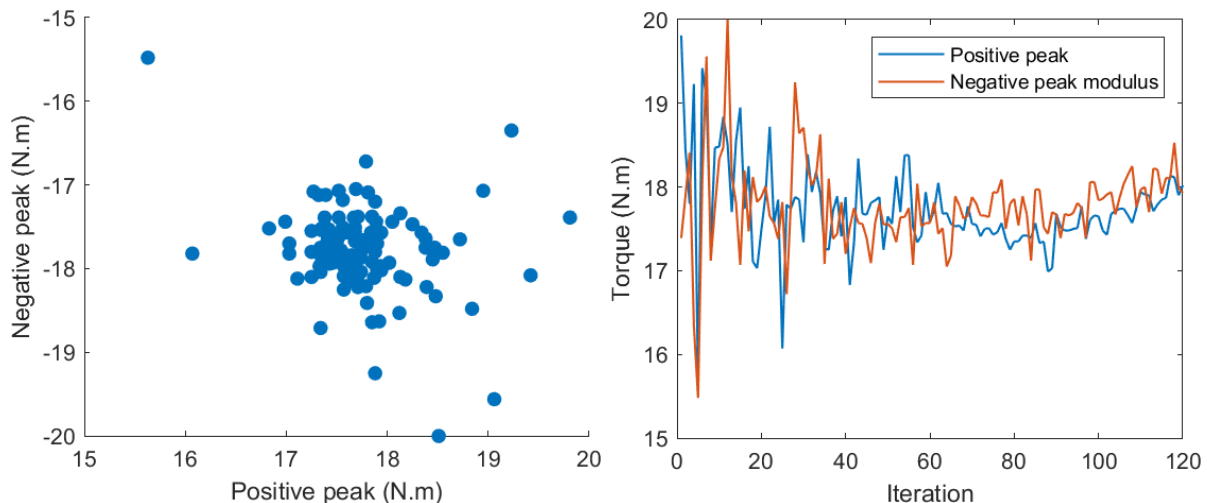
On the other hand, the torque profile's integral was given a smaller influence on the objective function's calculation. Despite its secondary role in the optimization, the presence of this variable in the optimization was imperative in guaranteeing a symmetric knee trajectory during the flexion and extension movements. However, a larger value of the ω_2^{OF} weight resulted in a failure to optimize the more critical physiological values that better relate to the users' exertion: the MC and interaction torque.

6.4.2 Human experiments

In this section, the results regarding the human experimental validation of the HITL controller are presented. Firstly, the results regarding the HITL optimization (first day of the experiment) are introduced. Later, are presented the results respecting the comparison of the optimized torque controller with the exoskeleton's zero-torque mode and walking without the SmartOs' device (second day of the experiment).

Human-in-the-loop optimization

Figure 45 presents the peak torque magnitudes for the various torque profiles generated in real-time during the HITL optimization process, which took a total of 20 minutes (120 CMA-ES iterations). Figure 45a presents all positive and negative peak combinations. On the other hand, Figure 45b shows the variation of both peaks' magnitude over the optimization time represented by the CMA-ES 120 iterations.



(a) Positive and negative peak combinations generated at the various iterations **(b)** Variation of the positive and negative peaks (in absolute) through the optimization process

Figure 45: Flexion (positive) and Extension (negative) torque magnitude values for the peaks of the various torque profiles tested during optimization.

From Figure 45a, it was observed that no torque magnitude peak reached the selected maximum or minimum value: 20 N.m and 15 N.m, for the positive peak, and -15 N.m and -20 N.m for the negative peak. Additionally, most of the peak values were between the $|17 \text{ N.m}|$ and $|18 \text{ N.m}|$. Furthermore, Figure 45 shows that the variation of the peak values was larger at the start of the optimization, and at this period both higher and lower peak values were tested. As the optimizer was running, both positive and negative peaks were getting closer and closer to the final solution and started fluctuating less and less. This was expected since as time passes the optimizer gets 'smarter' and nearer to the optimal solution.

Figure 46 depicts the variation of the objective function and its three parameters (MC, interaction torque, and reference torque's integral) over the optimization time represented by the optimizer's iterations. The objective function was calculated by Equation 23, using the weights obtained during the bench tests.

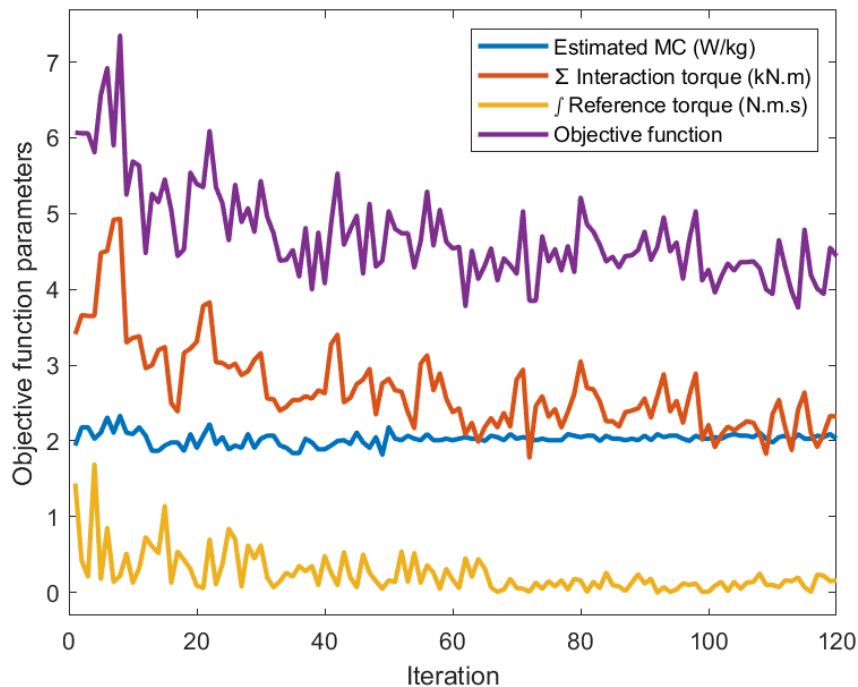


Figure 46: Evolution of the CMA-ES objective function and respective variables.

Figure 46 shows that both the interaction torque's cumulative sum and the reference torque's integral generally decrease with time. As previously explained, these two variables are related to some extent, as decreasing the reference's torque integral guarantees that the flexion and extension movement performed by the DC motor will be symmetric, and, therefore, the user will not need to 'add' extra torque to the system. However, the two curves were not perfectly proportional, and at some points, a small torque's integral generated a significant interaction torque. This can be explained by the motor failures identified in Chapter 5.4.2 that result in outlier interaction torque values.

Additionally, it was observed that the MC varies significantly at the start of the optimization process and gets more stable and constant during the second optimization half. This could be explained by the high torque variability at the start of the experiment, which resulted in abrupt movements performed by the participant. However, the MC does not significantly decrease during the second half of the optimization like the other two variables. This could be explained by the fact that the MC estimator solely focused on the acceleration data of the ankle, wrist, waist, and chest, which did not change significantly during the optimization second half, as the torque profile was getting closer to the optimal solution and its variability

was smaller. This caused the objective function to focus on the minimization of the interaction torque's cumulative sum, as seen by the similarity between these two curves, during the optimization's second half.

Figure 47 presents the optimal torque profile obtained by the CMA-ES optimizer after the 120 iterations. The torque magnitude of the positive curve's peak (flexion movement) was 18 N.m, while the magnitude of the negative peak (extension movement) was -17.9 N.m.

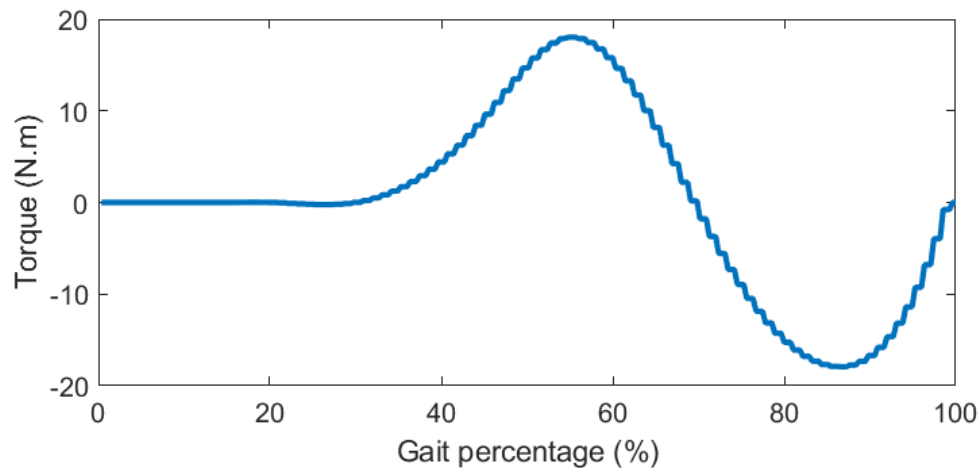


Figure 47: *Optimized torque profile.*

Figure 47 shows that the positive and negative curves of the optimal torque profile were quite balanced as the flexion and extension peaks were almost symmetrical. This was considered satisfactory as it guaranteed a small value for its integral (0.3 N.m.s). A narrow integral value, as previously explained, ensures that the SmartOs' motor fully performs both the flexion and extension movements. This leads to a much more comfortable gait pattern for the user.

Furthermore, the absolute value of the optimal peaks was roughly 18 N.m, an intermediate value between the defined maximum (20 N.m) and minimum (15 N.m). This agrees with the objective function definition, which tries to minimize both MC and interaction torque. Higher torques result in higher ankle acceleration values, and, therefore, higher estimated MC, and lower torques result in higher interaction torque since the user is forced to 'add' the extra torque necessary to perform the flexion and extension knee movements. Therefore, the objective function had to compromise and find a torque magnitude that was not too high nor too low.

Assistance comparison

Figure 48 presents a box plot of the participant's MC during the 5-minute protocol for each of the tested conditions. The red line at the center of each box plot represents the MC median. The first and

third quartiles are depicted by the bottom and top lines of the blue box, respectively, and the minimum and maximum MC by the limits of the whiskers (the black dotted lines outside each box).

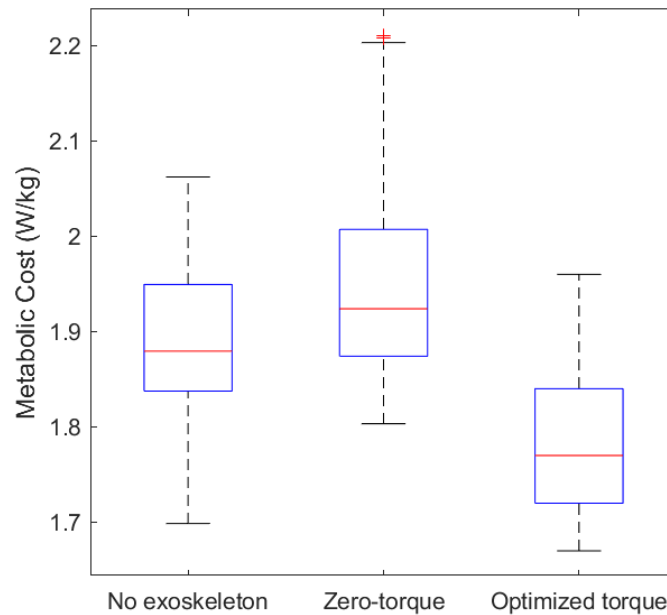


Figure 48: Boxplots depicting the MC of the participant during five minutes of walking at 1.5 km/h without the exoskeleton, and with the exoskeleton in zero-torque mode and with the optimized torque controller.

From the results depicted in Figure 48, it was possible to observe that, on average, the condition with lower MC was the optimized torque controller (the HITL strategy), demonstrating the efficacy of this controller. Additionally, it was observed that the condition with higher MC was the zero-torque control, which achieved MC values up to 2.2 W/kg. The higher MC during the zero-torque condition, compared to the no-exoskeleton condition, could be explained by the additional weight added to the user's right leg, which increased the user's effort, detected by the regression model due to an increase in the user's acceleration. These results are analogous to the conclusions in the literature studies, which obtained higher MC during the zero-torque conditions, and lower MC during the optimized control (8, 49).

Additionally, the average MC for the no-exoskeleton, the zero-torque, and the optimized torque conditions were 1.89 W/kg, 1.96 W/kg, and 1.79 W/kg, respectively. Therefore, the HITL-optimized controller achieved an average MC reduction of 5.3% and 8.7% when compared to the no-exoskeleton and zero-torque conditions, respectively. This reduction is lower than the results obtained in the literature (7% (57) up to 48% (53)), which could be explained by the fact that this work focused on the minimization of two physiological signals in simultaneous, the MC and interaction torque, in opposition to the literature studies, that only minimized the MC.

Furthermore, compared to the torque tracking controller presented in Chapter 5, which achieved an

average MC of 1.92 W/kg, the optimized controller accomplished an MC reduction of 6.8%. This corroborates with the discussion of the previous section, as it explains why the optimized torque magnitudes for the flexion and extension peaks were not set to the maximum allowed values (20 N.m and -20 N.m). Higher torque magnitudes result in faster movements of the lower limbs, which result in a higher estimated MC.

Figure 49 depicts the human-robot interaction torque during the experiment for each of the conditions where the participant wore the SmartOs (zero-torque mode and optimized torque controllers). Two red lines were used to represent the maximum and minimum values of the interaction torque measured. The interaction torque measured between minutes 2 and 2.5 (30-second period) is also presented to depict the signal during 12 gait cycles.

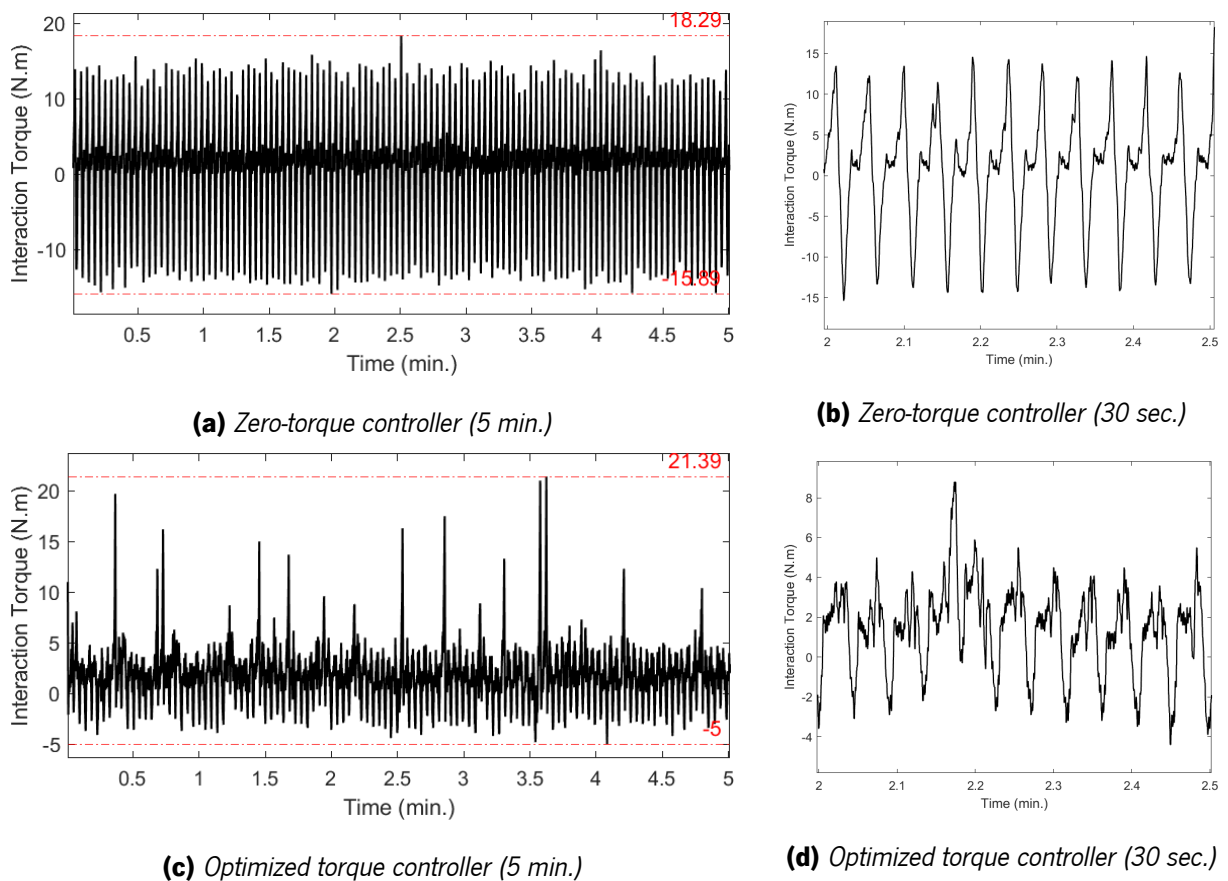


Figure 49: Interaction torque during five minutes (left) and 30 seconds (right) of walking at 1.5 km/h with a zero-torque controller (top) and the optimized torque controller (bottom). The red lines represent the maximum and minimum values of the interaction torque.

Figure 49 shows that the user's interaction torque was, generally, much superior during the experiment with the SmartOs' in zero-torque mode. This could be explained by the fact that, in the zero-torque mode, the person was in full command of the device and was forced to apply the necessary torque to drive it

in the desired motion. During the optimized torque strategy, the interaction torque was generally much lower since the exoskeleton was capable of performing most of the required torque to perform the desired knee movement. However, during the optimized torque control experiment, significant torque spikes were measured at random periods of the experiment. These outliers can be explained by the motor failures, already described in Chapter 5.4.2, that caused the participant to undergo high positive knee interaction torques to drive the motor to the desired position.

Additionally, the average of the absolute interaction torque was 5.27 N.m and 2.39 N.m for the zero-torque and the optimized torque conditions, respectively. Therefore, the optimized HITL controller achieved an interaction torque reduction of 54.6% on average, when compared to the zero-torque control.

Figure 50 presents the knee angle measured by SmartOs during the experiment, for the two controllers tested (the zero-torque controller and the optimized torque controller). The knee angle measured between minutes 2 and 2.5 (30-second period) is also presented to depict the signal during 12 gait cycles.

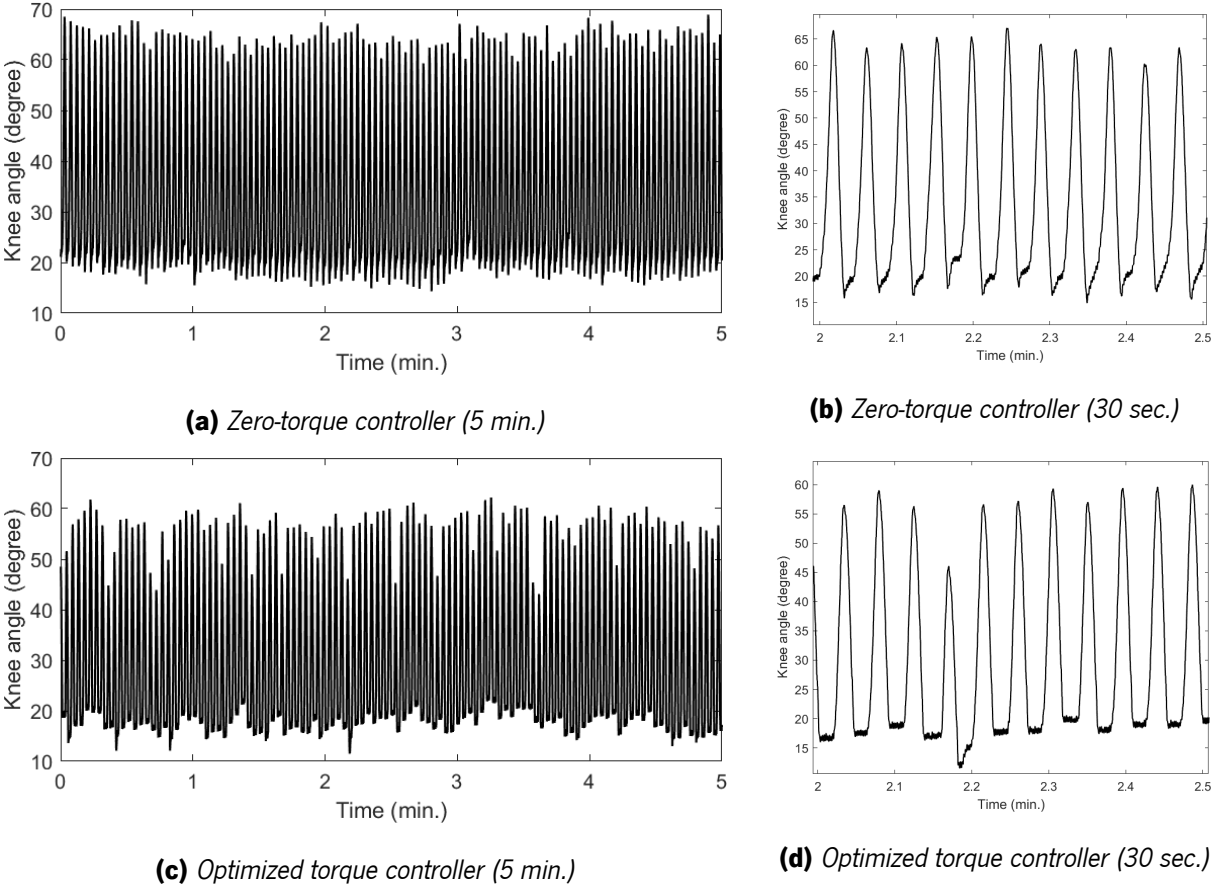


Figure 50: Knee angle during five minutes (left) and 30 seconds (right) of walking at 1.5 km/h with a zero-torque controller (top) and the optimized torque controller (bottom).

The results presented in Figure 50 depict that the participant had a more constant gait when using the zero-torque controller since the movement was entirely controlled by the person. The ROM when

walking with the zero-torque controller was roughly 55 degrees, while the ROM when walking with the optimized torque controller was roughly 45 degrees, 10 degrees lower than the first experiment. This smaller ROM could be explained by the relatively low torque magnitudes of the optimized torque profile (maximum of 18 N.m and minimum of -17.9 N.m) that resulted in a smaller variation of the knee's angle. However, this difference was not prejudicial to the participant's gait, who could still comfortably walk at the desired speed. The irregular knee movement at some points of the gait could also be explained by the DC motor failures since they caused sudden hiatuses to the knee's trajectory, suspending the movement momentarily.

6.5 Conclusions

This chapter presented the development of the HITL controller that was presented as the proposed solution to the problems identified in the state of the art. The controller successfully integrated the regression model for MC estimation presented in Chapter 4 and the torque tracking controller presented in Chapter 5 with a CMA-ES optimizer to minimize the exoskeleton user's MC and interaction torque in real-time. In conclusion, the HITL controller was successfully implemented into the SmartOs' architecture and proved to be effective in reducing both the user's MC and the interaction torque between the person and the exoskeleton and in guaranteeing a comfortable gait.

7 Conclusions

WMSDs are currently a major cause of concern among developed countries, affecting millions of European workers. It is estimated that three out of five workers suffer from these disorders, which have a higher prevalence in the hip and knee joints of the workers (29% and 33%, respectively). The industrial activities with a higher risk for WMSDs in the lower limbs are standing for long periods, and carrying and lifting heavy loads, as these activities comprise task repetitions, forceful exertions, and awkward postures. Despite the growing automation of industrial processes, humans are still needed to move heavy objects and perform repetitive tasks in harmful postures. Exoskeletons are portable devices worn by healthy workers to reduce the physical stress on the user's muscles and joints. These devices can, therefore, alleviate workers while they perform tasks that put them at risk for WMSDs, reducing the prevalence of these disorders.

LLEs have proven to be capable of reducing the user's physical exertion, however, various limitations to the employment of these devices can still be detected. One drawback of the current industrial exoskeletons is the lack of assistance strategies that adapt the device's movement to the user's demands in real-time (online optimization). One possible strategy is the optimization of LLE's control parameters based on users' physiological signals, usually the MC, denominated by HITL control. This approach has already proven to be effective in reducing the user's MC in laboratory settings, however, it has not yet been implemented in industrial applications. Additionally, measuring the MC of the user is not trivial, as the standard method for estimating this signal is through indirect calorimetry, which has countless limitations and is not feasible for HITL applications in real-world industrial backgrounds. A possible solution is the use of machine or deep learning models used to estimate the MC of a person based on data obtained from wearable sensors.

In this dissertation, an adaptive torque control was proposed for a knee exoskeleton (SmartOs), towards the power augmentation of workers during high-exertion tasks. The control employed a HITL strategy to optimize, in real-time, the torque profile of the LLE with the objective of reducing the user's exertion, measured by the MC and the human-robot interaction torque. Instead of estimating the MC through indirect calorimetry, this dissertation proposed the integration of a regression algorithm in the HITL control, capable of estimating this physiological signal based on a minimal number of wearable and non-intrusive sensors.

The first objective of this work was to conduct a literature review of LLEs, HITL controls, and regression algorithms for MC estimation. These reviews were presented in Chapter 2. The state of the art on industrial exoskeletons revealed the existence of two main LLE types: (i) devices for power augmentation in load-

carrying activities; and (ii) chair-like exoskeletons. Regarding the first type, most of the devices were actively actuated in at least one joint, more commonly the knee. In regards to the control strategies implemented in the analyzed exoskeletons, the devices were not designed to assist the users in an individualized approach nor considered the users' condition when wearing the device. The review of the HITL controllers showed that various optimization algorithms can be employed, with the most common method being a CMA-ES optimizer. Additionally, most of the literature studies optimized the users' MC, estimated by a respirometer device through indirect calorimetry, and took over 30 minutes of optimization time. The state of the art on regression models for MC estimation revealed that the studies varied significantly from each other regarding the activities performed, the model used, and the number, type, and location of the sensors. From this analysis, it was concluded that some of the best signals to estimate the MC were: (i) the HR; (ii) the waist, wrist, and ankle acceleration; and (iii) the user's weight and gender. The KPI for this objective was the production of the review, and was, therefore, considered concluded.

In Chapter 4, a regression model for MC estimation was developed and integrated into SmartOs' architecture. The model was able to estimate the MC of a person, in real-time and in 10-second intervals, based on data acquired from four wearable and non-intrusive inertial sensors (3D acceleration) located at the chest, left waist, right wrist, and right ankle. This procedure covered various steps: (i) the selection of a public dataset with relevant data; (ii) the optimization of the data preprocessing method; (iii) the sensors selection for the MC estimation, without disregarding their feasibility for HITL applications in industrial contexts; (iv) the development and comparison of various regression models, using the LOOCV technique; (v) the implementation of the model into the SmartOs' architecture; and (vi) the validation of the model in a real-time human experimental. The model was successfully developed and integrated into the exoskeleton system, with low computational cost and without latency. An offline test showed that the model achieved an RMSE of 0.45 W/kg and a coefficient of determination (R^2) of 0.84, better values than the metrics established as the KPI of the model (0.8 W/kg and 0.8, respectively). The validation in real-time obtained an RMSE 46.7% higher than the offline test (0.66 W/kg), however, these results were still lower than the KPI, and the model was considered effective. Additionally, this work allowed a reduction in the time required to estimate the steady-state MC from 3 minutes to 10 seconds, when compared to the indirect calorimetry method. Therefore, considering these results, and the fact that the model only required the data from four wearable and non-intrusive sensors, the algorithm was considered suitable to be integrated into a HITL control.

In Chapter 5, a torque tracking control for the knee exoskeleton was developed. This control was able to drive the LLE's motor by following a desired knee torque profile, and was composed of three main blocks:

(i) a natural cubic interpolator used to generate any desired torque profile; (ii) a mid-level controller used to estimate the gait cycle phase and generate the reference torque from the torque profile and the estimated gait cycle phase; and (iii) a low-level controller used to drive the motor in the desired torque trajectory, by minimizing the difference between the reference torque and the actuator's real torque. This was followed by the identification of the ideal torque profile shape for an electric motor, which was composed of one flexion and one extension curve, with peaks at 55% and 86% of the gait cycle. A conceptual experimental validation of the torque tracking control showed that the control was able to follow the desired torque profile and that this torque profile enabled the user to walk comfortably, without constraints. The user's movement was considered continuous and regular, and the control allowed a low human-device interaction torque during the experiment (-4 N.m to 7 N.m). Therefore, the control achieved its established KPI, which was ensuring a comfortable gait, and was considered adequate to be integrated into the HITL control proposed in this dissertation.

In Chapter 6, a HITL adaptive torque control was developed, by integrating the regression model for MC estimation and the torque tracking control with a CMA-ES optimizer. The HITL control was capable of adapting the torque profile provided to the torque tracking control, in real-time, by minimizing the user's exertion. The exertion was measured by the user's estimated MC (through the regression model) and the user's interaction torque. Two control parameters were optimized: the torque magnitude of the flexion and extension peaks of the knee torque profile. The CMA-ES optimizer was capable of finding an optimal solution in 20 generations, a total of 20 minutes of optimization time, a value lower than the established KPI for this control (30 minutes). Finally, a conceptual experimental validation of the HITL control was performed. The optimized torque control achieved an 8.7% reduction of the user's MC when compared to a zero-torque condition, a value slightly lower than the KPI (10%). Furthermore, the control achieved a 54.6% reduction of the user's interaction torque, which in this case was a value significantly higher than the KPI (10%). The control was successfully integrated into SmartOs' architecture, running smoothly and without any latency, and was considered effective in minimizing the exoskeleton user's exertion.

The goals proposed for this dissertation (presented in Chapter 1.3) were therefore considered fulfilled. Furthermore, the work here presented was able to answer the research questions established in Chapter 1.4:

- **RQ1:** What are the current effects of HITL controls implemented on LLEs?

The answer to this question can be found in Chapter 2. From the state of the art analysis on HITL controls, it was possible to observe that all controls were effective in optimizing their desired physiological signal (the optimized signal varied from study to study). Most of the controls adapted

the control parameters of an exoskeleton based on the user's MC. These studies achieved MC reductions from 7% to 48%, obtaining, therefore, almost half of the MC when using optimized assistance to each user. Studies that minimized the user's muscle activity achieved reductions of this signal of 21%. Furthermore, studies that optimized the user's gait speed were able to reduce this parameter by 42%. All these results were obtained when comparing the optimized control with a no-exoskeleton condition, in an experimental protocol.

- **RQ2:** How accurate can a regression model be in estimating the MC based on a small number of wearable and non-intrusive sensors?

The answer to this question can be found in Chapter 4. The MC estimating regression model developed in this dissertation passed through two main validation stages: the offline validation/test, using the LOOCV method where one random subject was chosen for testing the model; and the real-time validation, where various participants performed an experimental protocol and the estimation made by the model was compared to the estimation made through indirect calorimetry. During the offline validation, the regression model achieved an RMSE and a coefficient of determination (R^2) of 0.31 W/kg and 0.91, respectively. Regarding the offline test, the regression model achieved an average RMSE of 0.45 W/kg and an R^2 of 0.84. As to the real-time experimental validation, the average RMSE between all participants was 0.66 W/kg, while the average R^2 was 0.75.

- **RQ3:** How much can LLE users' exertion be reduced by a real-time HITL optimization?

The answer to this question can be found in Chapter 6. The HITL control presented in this dissertation was used to optimize a torque profile of a knee exoskeleton, in real-time, by minimizing the user's exertion, evaluated by a weighted sum of the user's MC and interaction torque. The effectiveness of the control was assessed by comparing the user's exertion when walking without the exoskeleton, and when walking with the device in a zero-torque mode and with the optimized torque control. The optimized assistance achieved an MC reduction of 5.3 % and 8.7% compared to the no-exoskeleton and zero-torque conditions, respectively. Additionally, the HITL control enabled interaction torque reductions of 54.6% compared to walking with the exoskeleton in the zero-torque mode.

7.1 Future work

Future work to extend the work developed in this dissertation comprises of: (i) the integration of an HR monitor sensor into the SmartOs system, to allow the MC estimation by a regression model based not only

on the 3D acceleration data but also on the HR signal measured in real-time; (ii) the development of a new regression model capable of estimating the MC during a large set of activities, especially activities with higher industrial relevance, such as lifting and carrying heavy loads; and (iii) the experimental validation of the HITL control with more participants.

References

- (1) de Kok J., Vroonhof P., Snijders J., Roullis G., Clarke M., Peereboom K., van Dorst P. and Isusi I. (2019). Work-related musculoskeletal disorders: prevalence, costs and demographics in the EU.
- (2) Chen B., Ma H., Qin L., Gao F., Chan K., Law S., Qin L. and Liao W. (2016). Recent developments and challenges of lower extremity exoskeletons. *Journal of Orthopaedic Translation* 5, 26–37.
- (3) World Health Organization Musculoskeletal health, <https://www.who.int/news-room/fact-sheets/detail/musculoskeletal-conditions>, [accessed on 2022-12-27], 2022.
- (4) Govaerts R., Tassignon B., Ghillebert J., Serrien B., De Bock S., Ampe T., El Makrini I., Vanderborght B., Meeusen R. and De Pauw K. (2021). Prevalence and incidence of work-related musculoskeletal disorders in secondary industries of 21st century Europe: a systematic review and meta-analysis. *BMC Musculoskeletal Disorders* 22:751.
- (5) Eurostat Accidents at work - statistics on causes and circumstances, https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Accidents_at_work_-_statistics_on_causes_and_circumstances&oldid=579777%E2%80%A6, [accessed on 2022-12-14], 2020.
- (6) ErgoPlus The Definition and Causes of Musculoskeletal Disorders, <https://ergo-plus.com/musculoskeletal-disorders-msd/>, [accessed on 2022-11-23].
- (7) Mishra S. and Sarkar K. (2021). Work-related musculoskeletal disorders and associated risk factors among urban metropolitan hairdressers in India. *Journal of Occupational Health* 63:e12200.
- (8) Zhang J., Fiers P., Witte K.A., Jackson R.W., Poggensee K.L., Atkenson C.G. and Collins S.H. (2017). Human in the loop optimization of exoskeleton assistance during walking. *Science* 356:6344, 1280–1283.
- (9) Zoss A. B., Kazerooni H. and Chu A. (2006). Biomechanical Design of the Berkeley Lower Extremity Exoskeleton (BLEEX). *IEEE/ASME Transactions On Mechatronics* 11:2, 128–138.
- (10) Walsh C. J., Endo K. and Herr H. (2007). A quasi-passive leg exoskeleton for load-carrying augmentation. *International Journal of Humanoid Robotics* 4:3, 487–506.

- (11) Kim M., Ding Y., Malcolm P., Speeckaert J., Siviý C.J., Walsh C.J., et al. (2017). Human-in-the-loop Bayesian optimization of wearable device parameters. *PLoS ONE* 12:9.
- (12) Song, S. and Collins S.H. (2021). Optimizing Exoskeleton Assistance for Faster Self-Selected Walking. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 29, 786–795.
- (13) Koller J., Gates D., Ferris D., Remy C.D., and Arbor A. (2016). 'Body-in-the-Loop' Optimization of Assistive Robotic Devices: A Validation Study. *Robotics: Science and Systems* 12.
- (14) Slade P., Troutman R., Kochenderfer M.J., Collins S.H., Delp S.L. (2019). Rapid energy expenditure estimation for ankle assisted and inclined loaded walking. *Journal of NeuroEngineering and Rehabilitation* 16:67.
- (15) Slade P., Kochenderfer M.J., Delp S.L. and Collins S.H. (2021). Sensing leg movement enhances wearable monitoring of energy expenditure. *Nature Communications* 12.
- (16) Lopes J.M., Figueiredo J., Fonseca P., Cerqueira J.J., Vilas-Boas J.P. and Santos C.P. (2022). Deep Learning-Based Energy Expenditure Estimation in Assisted and Non-Assisted Gait Using Inertial, EMG, and Heart Rate Wearable Sensors. *Sensors* 22.
- (17) Gjoreski H., Kaluža B., Matjaž G., Milić and Luštrek M. (2015). Context-based ensemble method for human energy expenditure estimation. *Applied Soft Computing* 37, 960–970.
- (18) Zhu J., Pande A., Mohapatra P. and Han J.J. (2015). Using Deep Learning for Energy Expenditure Estimation with Wearable Sensors. *17th International Conference on E-health Networking, Application Services (HealthCom)* 16:67.
- (19) Ding Y., Kim M., Kuindersma S. and Walsh C.J. (2018). Human-in-the-loop optimization of hip assistance with a soft exosuit during walking. *Science Robotics* 3:15.
- (20) de Looze M. P., Bosch T., Krause F., Stadler K. S. and O'Sullivan L. W. (2016). Exoskeletons for industrial application and their potential effects on physical work load. *Ergonomics* 59:5, 671–681.
- (21) Bär M., Steinhilber B., Rieger M. A., Luger T. (2021). The influence of using exoskeletons during occupational tasks on acute physical stress and strain compared to no exoskeleton – A systematic review and meta-analysis. *Applied Ergonomics* 94.
- (22) Herr H. (2009). Exoskeletons and orthoses: classification, design challenges and future directions. *Journal of NeuroEngineering and Rehabilitation* 6.
- (23) Wevolver BLEEX, <https://www.wevolver.com/specs/bleex>, [accessed on 2023-09-07].

- (24) IMECHE Rise of the cobots, <https://www.imeche.org/news/news-article/rise-of-the-cobots>, [accessed on 2023-09-07].
- (25) de zeen "Chairless chair" is designed to provide support for active factory workers, <https://www.dezeen.com/2017/07/06/chairless-chair-designed-provide-support-active-factory-workers/>, [accessed on 2023-09-07].
- (26) Pinterest "LegX - Knee Exoskeleton that adapts and boost your strength", <https://br.pinterest.com/pin/313774299037670435/>, [accessed on 2023-09-07].
- (27) Capitani S. L., Bianchi M., Secciani N., Pagliai M., Meli E. and Ridolfi A. (2021). Model-based mechanical design of a passive lower-limb exoskeleton for assisting workers in shotcrete projection. *Meccanica* 56, 195–210.
- (28) Pratt J. E., Collins S. H., Krupp B. T. and Morse C. J. (2004). The RoboKnee: An Exoskeleton for Enhancing Strength and Endurance During Walking. *IEEE International Conference on Robotics Automation, New Orleans, LA*.
- (29) Low K. H., Lu X. and Yu H. (2005). Development of NTU Wearable Exoskeleton System for Assistive Technologies. *IEEE International Conference on Mechatronics Automation, Niagara Falls, Canada*.
- (30) Army Technology Human Universal Load Carrier (HULC), <https://www.army-technology.com/projects/human-universal-load-carrier-hulc/>, [accessed on 2022-10-27].
- (31) Lim D., Lim W., Lee H., Kim H., Shin K., Park T., Lee J. and Han C. (2015). Development of a Lower Extremity Exoskeleton Robot with a Quasi-anthropomorphic Design Approach for Load Carriage. *IEEE/RSJ International Conference on Intelligent Robots and Systems, Hamburg, Germany*.
- (32) Direct Industry Mechanical exoskeleton Hercule V3, <https://www.directindustry.com/prod/rb3d/product-180292-2299769.html>, [accessed on 2022-10-27].
- (33) Direct Industry EXOSKELETONS HERCULE, https://pdf.directindustry.com/pdf/rb3d/exoskeletons-hercule/180292-673504-_2.html, [accessed on 2022-10-27].
- (34) The Verge Hercule exoskeleton can help a regular human carry up to 220 pounds, <https://www.theverge.com/2012/2/22/2815704/hercule-exoskeleton-human-carry-220-pounds>, [accessed on 2022-10-27].

- (35) Sado F., Yap H. J., Ghazilla R. and Ahmad N. (2019). Design and control of a wearable lower-body exoskeleton for squatting and walking assistance in manual handling works. *Mechatronics* 63, 195–210.
- (36) TWI WHAT ARE TECHNOLOGY READINESS LEVELS (TRL)?, <https://www.twi-global.com/technical-knowledge/faqs/technology-readiness-levels>, [accessed on 2023-09-07].
- (37) Noonee Chairless Chair 2.0 – The new generation, <https://www.noonee.com/?lang=en>, [accessed on 2022-10-05].
- (38) Archelis Product introduction, <https://en.archelis.com/product-factory>, [accessed on 2022-10-05].
- (39) Pillai M. V., Engelhoven L. V. and Kazerooni H. (2020). Evaluation of a Lower Leg Support Exoskeleton on Floor and Below Hip Height Panel Work. *Human Factors* 62:3, 489–500.
- (40) Wijegunawardana I., Kumara M., De Silva H., Viduranga P., Ranaweera R., Gopura R. and Madusanka D. (2019). ChairX: A Robotic Exoskeleton Chair for Industrial Workers. *IEEE International Conference on Rehabilitation Robotics*, 587–592.
- (41) Tu Y., Zhu A., Song J., Zhang X. and Cao G. (2022). Design and Experimental Evaluation of a Lower-Limb Exoskeleton for Assisting Workers With Motorized Tuning of Squat Heights. *IEEE Transactions On Neural Systems and Rehabilitation Engineering* 30, 184–193.
- (42) Kim H.J., Noh J. and Yang W. (2020). Knee-assistive robotic exoskeleton (KARE-1) using a conditionally singular mechanism for industrial field applications. *Applied Sciences (Switzerland)* 10:15.
- (43) Kong Y., Park C., Cho M., Kim S., Kim S., Kim M., Hyun D., Bae K., Choi J., Ko S. and Choi K. (2021). Guidelines for Working Heights of the Lower-Limb Exoskeleton (CEX) Based on Ergonomic Evaluations. *International Journal of Environmental Research and Public Health* 18:10.
- (44) Baud R., Manzoori A.R., Ijspeert A. and Bouri M. (2021). Review of control strategies for lower-limb exoskeletons to assist gait. *Journal of NeuroEngineering and Rehabilitation* 18:1.
- (45) Young A.J. and Ferris D. P. (2017). State of the Art and Future Directions for Lower Limb Robotic Exoskeletons. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25:2, 171–182.

- (46) FDA (2013). Evaluation of Automatic Class III Designations (De Novo) For ARGO ReWalk™. *21 CFR 890.3480*.
- (47) He Y., Eguren D., Luu T.P. and Contreras-Vidal J.L. (2017). Risk management and regulations for lower limb medical exoskeletons: A review. *Medical Devices: Evidence and Research 10*, 89–107.
- (48) Rodríguez-Fernández A., Lobo-Prat J. and Font-Llagunes J.M. (2021). Systematic review on wearable lower-limb exoskeletons for gait training in neuromuscular impairments. *Journal of NeuroEngineering and Rehabilitation 18:1*.
- (49) Witte K.A., Fiers P., Sheets-Singer A.L. and Collins S.H. (2020). Improving the energy economy of human running with powered and unpowered ankle exoskeleton assistance. *Science Robotics 5:40*.
- (50) Han H., Wang W., Zhang F., Li X., Chen J., Han J., Zhang J. (2021). Selection of Muscle-Activity-Based Cost Function in Human-in-the-Loop Optimization of Multi-Gait Ankle Exoskeleton Assistance. *IEEE Transactions on Neural Systems and Rehabilitation Engineering 29*, 944–952.
- (51) Bryan G.M., Franks P.W., Song S., Reyes R., O'Donovan M.P., Gregorczyk K.N. and Collins S.H. (2021). Optimized hip-knee-ankle exoskeleton assistance reduces the metabolic cost of walking with worn loads. *Journal of NeuroEngineering and Rehabilitation 18:1*.
- (52) Bryan G.M., Franks P.W., Song S., Voloshina A.S., Reyes R., O'Donovan M.P., Gregorczyk K.N. and Collins S.H. (2021). Optimized hip–knee–ankle exoskeleton assistance at a range of walking speeds. *Journal of NeuroEngineering and Rehabilitation 18:1*.
- (53) Franks P.W., Bryan G.M., Reyes R., O'Donovan M.P., Gregorczyk K.N. and Collins S.H. (2022). The Effects of Incline Level on Optimized Lower-Limb Exoskeleton Assistance: a Case Series. *IEEE Transactions on Neural Systems and Rehabilitation Engineering 30*, 2494–2505.
- (54) Kim J., Quinlivan B.T., Deprey L., Revi D.A., Eckert-Erdheim A., Murphy P., Orzel D. and Walsh C.J. (2022). Reducing the energy cost of walking with low assistance levels through optimized hip flexion assistance from a soft exosuit. *Scientific Reports 12:1*.
- (55) Xu L., Liu X., Chen Y., Yu L., Yan Z., Yang C., Zhou C. and Yang W. (2023). Reducing the muscle activity of walking using a portable hip exoskeleton based on human-in-the-loop optimization. *Frontiers in Bioengineering and Biotechnology 11*.

- (56) Kantharaju P., Jeong H., Ramadurai S., Jacobson M., Jeong H. and Kim M. (2022). Reducing Squat Physical Effort Using Personalized Assistance From an Ankle Exoskeleton. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 30, 1786–1795.
- (57) Kim M., Liu C., Kim J., Lee S., Meguid A., Walsh C. and Kuindersma S. (2019). Bayesian Optimization of Soft Exosuits Using a Metabolic Estimator Stopping Process. *IEEE International Conference on Robotics and Automation*, 9173–9179.
- (58) Gordon D.F.N., McGreavy C., Christou A. and Vijayakumar S. (2022). Human-in-the-Loop Optimization of Exoskeleton Assistance Via Online Simulation of Metabolic Cost. *IEEE Transactions on Robotics* 38:3, 1410–1429.
- (59) Jackson R.W. and Collins S.H. (2019). Heuristic-Based Ankle Exoskeleton Control for Co-Adaptive Assistance of Human Locomotion. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 27:10, 2059–2069.
- (60) Yan H., Tang B., Xiang K. and Pang M. (2019). Human-in-the-loop optimization control for the ankle exoskeleton during walking based on iterative learning and particle swarm optimization algorithm. *IEEE International Conference on Advanced Robotics and Mechatronics, ICARM*, 570–574.
- (61) Tucker M., Myra C., Novoseller E., Cheng R., Yue T., Burdick J.W. and Ames A.D. (2020). Human Preference-Based Learning for High-dimensional Optimization of Exoskeleton Walking Gaits. *IEEE International Conference on Intelligent Robots and Systems*, 3423–3430.
- (62) Slade P., Kochenderfer M.J., Delp S.L., Collins S.H. (2022). Personalizing exoskeleton assistance while walking in the real world. *Nature* 610, 277–282.
- (63) Alvarez-Garcia J.A., Cvetković B. and Lustrek M. (2020). A Survey on Energy Expenditure Estimation Using Wearable Devices. *ACM Computing Surveys* 53:5, 1–35.
- (64) Silder A., Besier T. and Delp S.L. (2012). Predicting the metabolic cost of incline walking from muscle activity and walking mechanics. *Journal of Biomechanics* 45, 1842–1849.
- (65) Strath S.J., Kate R.J., Keenan K.G., Welch W.A. and Swartz A.M. (2015). Ngram time series model to predict activity type and energy cost from wrist, hip and ankle accelerometers: implications of age. *Physiological Measurement* 36, 2335–2351.
- (66) Ingraham K.A., Ferris D.P. and Remy C.D. (2019). Evaluating physiological signal salience for estimating metabolic energy cost from wearable sensors. *Journal of Applied Physiology* 126, 717–729.

- (67) Pande A., Zhu J., Das A.K., Zeng Y., Mohapatra P. and Han J.J. (2015). Using Smartphone Sensors for Improving Energy Expenditure Estimation. *IEEE Journal of Translational Engineering in Health and Medicine* 3.
- (68) Sazonov E., Hegde N., Browning R.C., Melanson E.L. and Sazonova N.A. (2015). Posture and Activity Recognition and Energy Expenditure Estimation in a Wearable Platform. *IEEE Journal of Biomedical and Health Informatics* 19.
- (69) Catal C. and Akbulut A. (2018). Automatic energy expenditure measurement for health science. *Computer Methods and Programs in Biomedicine* 157, 31–37.
- (70) Aziz O., Zihajehzadeh S., Park A., Tae C. and Park E. (2020). Improving Energy Expenditure Estimation through Activity Classification and Walking Speed Estimation Using a Smartwatch. *IEEE Journal of Translational Engineering in Health and Medicine*, 3940–3944.
- (71) Sevil M., Rashid M., Maloney Z., Hajizadeh I., Samadi S., Askari M.R., Hobbs N., Brandt R., Park M. and Quinn L. (2020). Determining Physical Activity Characteristics From Wristband Data for Use in Automated Insulin Delivery Systems. *IEEE Sensors Journal* 20, 12859–12870.
- (72) Lucena A., Guedes J., Vaz M., Silva L., Bustos D. and Souza E. (2021). Modeling Energy Expenditure Estimation in Occupational Context by Actigraphy: A Multi Regression Mixed-Effects Model. *International Journal of Environmental Research and Public Health* 18.
- (73) Ni Z., Wu T., Wang T., Su F. and Li Y. (2022). Deep Multi-Branch Two-Stage Regression Network for Accurate Energy Expenditure Estimation With ECG and IMU Data. *IEEE Transactions on Biomedical Engineering* 69, 3224–3233.
- (74) Ramadurai S., Jeong H. and Kim M. (2023). Predicting the metabolic cost of exoskeleton-assisted squatting using foot pressure features and machine learning. *Frontiers in Robotics and AI* 10.
- (75) The Guardian Tories have unhealthy financial reliance on property developers, says report, <https://www.theguardian.com/society/2021/jul/12/tories-have-unhealthy-financial-reliance-on-property-developers-says-report>, [accessed on 2023-09-07].
- (76) The Guardian Tories have unhealthy financial reliance on property developers, says report, <https://www.theguardian.com/society/2021/jul/12/tories-have-unhealthy-financial-reliance-on-property-developers-says-report>, [accessed on 2023-09-07].

- (77) Shutterstock, https://www.shutterstock.com/pt/image-photo/young-bearded-worker-technical-service-sitting-1517895674?consentChanged=true&irclickid=xGq3NewfoxyPWI3zghSkRSE8UkFU0lyjxUtA3c0&irgwc=1&utm_campaign=TinEye&utm_content=108110&utm_medium=Affiliate&utm_source=77643&utm_term=, [accessed on 2023-09-07].
- (78) how stuff works Who Invented the Assembly Line?, <https://science.howstuffworks.com/innovation/inventions/who-invented-the-assembly-line.htm>, [accessed on 2023-09-07].
- (79) Han Y. and Wang X. (2011). The biomechanical study of lower limb during human walking. *Science China Technological Sciences* 54:4, 83–991.
- (80) Hwang S., Kim Y. and Kim Y. (2009). Lower extremity joint kinetics and lumbar curvature during squat and stoop lifting. *BMC Musculoskeletal Disorders* 10.
- (81) Reznick E., Embry K.R., Neuman R., Bolívar-Nieto E., Fey N.P. and Gregg R.D. (2021). Lower-limb kinematics and kinetics during continuously varying human locomotion. *Scientific Data* 10:1.
- (82) Pinheiro C., Lopes J.M., Moreira L., Sanz-Merodio D., Figueiredo J., Santos C.P. and Garcia E. (2019). Kinematic and kinetic study of sit-to-stand and stand-to-sit movements towards a human-like skeletal model. *2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG)* 18613936.
- (83) Figueiredo, J. Smart Wearable Orthosis to Assist Impaired Human Walking, Available at <https://repositorium.sdum.uminho.pt/handle/1822/65877>, PhD thesis, University of Minho, 2019.
- (84) Figueiredo J., Carvalho S.P., Vilas-Boas J.P., Gonçalves L.M., Moreno J.C. and Santos, C.P. (2020). Wearable Inertial Sensor System towards Daily Human Kinematic Gait Analysis: Benchmarking Analysis to MVN BIOMECH. *Sensors (Switzerland)* 20:8.
- (85) Brockway J. (1987). Derivation of formulae used to calculate energy expenditure in man. *Hum Nutr Clin Nutr* 41:6, 463–471.
- (86) Wikipediaybt Anatomical Plane, https://en.wikipedia.org/wiki/Anatomical_plane, [accessed on 2023-09-07].
- (87) Adeyeri B., Thomas S.A., and Arellano C.J. (2022). A simple method reveals minimum time required to quantify steady-rate metabolism and net cost of transport for human walking. *Journal of Experimental Biology* 225:15.

- (88) Cvetković B., Milić R. and Luštrek M. (2016). Estimating Energy Expenditure with Multiple Models Using Different Wearable Sensors. *IEEE Journal of Biomedical and Health Informatics* 20, 1081–1087.
- (89) Gashi S., Min C., Montanari A., Santini S. and Kawsar F. (2022). A multidevice and multimodal dataset for human energy expenditure estimation using wearable devices. *Scientific Data* 9:1.
- (90) Ingraham's dataset, https://figshare.com/articles/dataset/Predicting_energy_cost_from_wearable_sensors_A_dataset_of_energetic_and_physiological_wearable_sensor_data_from_healthy_individuals_performing_multiple_physical_activities/7473191, [accessed on January 2023].
- (91) Su S.W., Wang L., Celler B.G., Ambikairajag E., and Savkin A.V. (2015). Estimation of Walking Energy Expenditure by Using Support Vector Regression. *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, 1–4.
- (92) Bazuelo-Ruiz B., De Rosario H., and Durá-Gil J.V. (2022). Estimation of energy expenditure in adults with accelerometry and heart rate. *Science Sports* 37, 431–437.
- (93) Miguel-Fernández J., Pescatore C., Mesa-Garrido A., Rikhof C., Prinsen E., Font-Llagunes J.M., and Lobo-Prat J. (2022). Immediate Biomechanical Effects of Providing Adaptive Assistance With an Ankle Exoskeleton in Individuals After Stroke. *IEEE ROBOTICS AND AUTOMATION LETTERS* 7:3.
- (94) Miguel-Fernández J., Mesa-Garrido A., Pescatore C., Rikhof C., Lobo-Prat J., Font-Llagunes J.M. (2022). Relationship Between Ankle Assistive Torque and Biomechanical Gait Metrics in Individuals After Stroke. *TechRxiv*.
- (95) Grimmer M. and Seyfarth A. (2014). Mimicking Human-Like Leg Function in Prosthetic Limbs. *Trends in Augmentation of Human Performance, Chapter 5, Springer Netherlands*.
- (96) Burden R.L. and Faires J.D., *Numerical Analysis*, 9th ed.; Cengage Learning: 2010, pp 144–153.
- (97) Ellis G., *Control System Design Guide*, 4th ed.; Butterworth-Heinemann: 2012, pp 108–110.
- (98) Ingraham K.A., Rouse E.J and Remy C.D (2020). Accelerating the Estimation of Metabolic Cost Using Signal Derivatives. *IEEE Robotics Automation Magazine* 27:1, 32–42.
- (99) Hansen, N. (2006). The CMA Evolution Strategy: A Comparing Review. *Studies in Fuzziness and Soft Computing* 192, 75–102.

- (100) Hansen, N. (2005). The CMA Evolution Strategy: A Tutorial. *hal-01297037v2*.
- (101) Repository with the base CMA-ES code in C, <https://github.com/CMA-ES/c-cmaes>, [accessed on April 2023].

A Appendix: Cross-correlation between Ingraham's and InertiaLab's 3D acceleration axes

Tables 26, 27, 28, and 29 present the results of the cross-correlation study performed for Ingraham's and InertiaLab's 3D acceleration axes, for the right wrist, chest, left waist, and right ankle IMUs, respectively. The best correspondence between the two systems is marked in red.

Table 26: Wrist axis cross-correlation between Ingraham's and InertiaLab's data. Legend: IL - "InertiaLab"

	IL's x-axis	IL's y-axis	IL's z-axis	Inverted IL's x-axis	Inverted IL's y-axis	Inverted IL's z-axis
Ingraham's x-axis	-0.2	140.9	0	399.5	0	86.3
Ingraham's y-axis	-0.6	394	-0.1	1109.6	-0.2	240.6
Ingraham's z-axis	145.8	0	32.4	0	52.4	0

Table 27: Chest axis cross-correlation between Ingraham's and InertiaLab's data. Legend: IL - "InertiaLab"

	IL's x-axis	IL's y-axis	IL's z-axis	Inverted IL's x-axis	Inverted IL's y-axis	Inverted IL's z-axis
Ingraham's x-axis	0	0	564.7	4960.3	283.6	0
Ingraham's y-axis	0	0	119.6	1033.3	60.9	0
Ingraham's z-axis	802.4	46.6	0	0.1	0	91.4

Table 28: Waist axis cross-correlation between Ingraham's and InertiaLab's data. Legend: IL - "InertiaLab"

	IL's x-axis	IL's y-axis	IL's z-axis	Inverted IL's x-axis	Inverted IL's y-axis	Inverted IL's z-axis
Ingraham's x-axis	0	262.8	690.2	5224	2.5	0
Ingraham's y-axis	0	44.2	107.8	797	0.6	0
Ingraham's z-axis	0	41.8	101.7	753.3	0.6	0

Table 29: Ankle axis cross-correlation between Ingraham's and InertiaLab's data. Legend: IL - "InertiaLab"

	IL's x-axis	IL's y-axis	IL's z-axis	Inverted IL's x-axis	Inverted IL's y-axis	Inverted IL's z-axis
Ingraham's x-axis	0	698.7	1734.8	6475.4	47.1	0
Ingraham's y-axis	10.7	126.8	168.5	503.4	93.7	20.7
Ingraham's z-axis	0	180.9	369.4	1317	22	0.5

B Appendix: 3D acceleration acquired during the MC estimation validation

Figures 51, 52, 54, 56, 58, and 60 present the acceleration values, along each axis, of Participants 1, 2 (1st and 2nd trials), 3, 4, and 5, respectively, respective to the signals that were used to estimate the MC (acceleration of the chest, right wrist, left waist, and right ankle). Figures 53, 55, 57, 59, and 61 present the acceleration of the other locations (right waist, back waist, and right knee), along each axis, of Participants 2 (1st and 2nd trials), 3, 4, and 5, respectively.

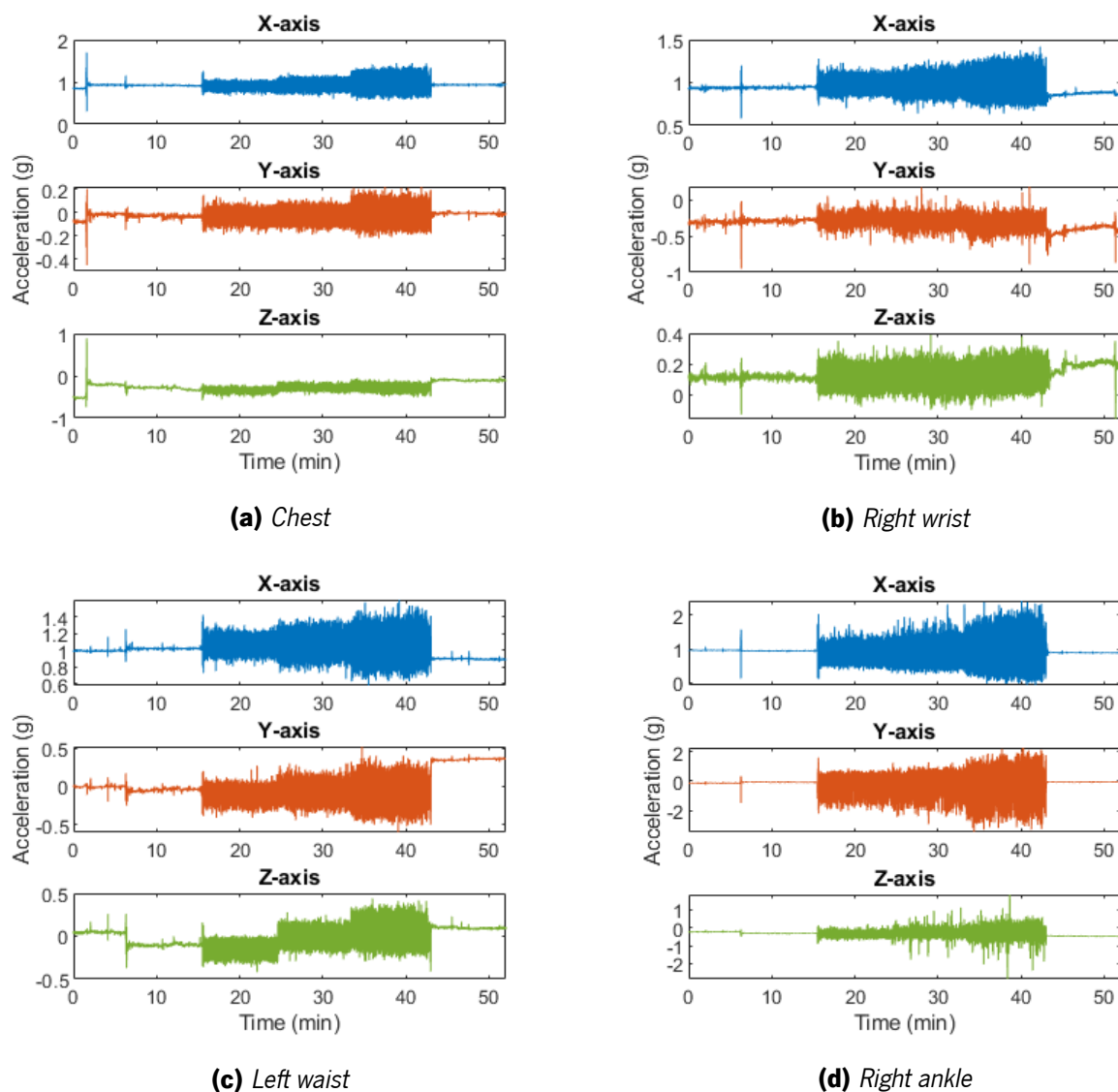
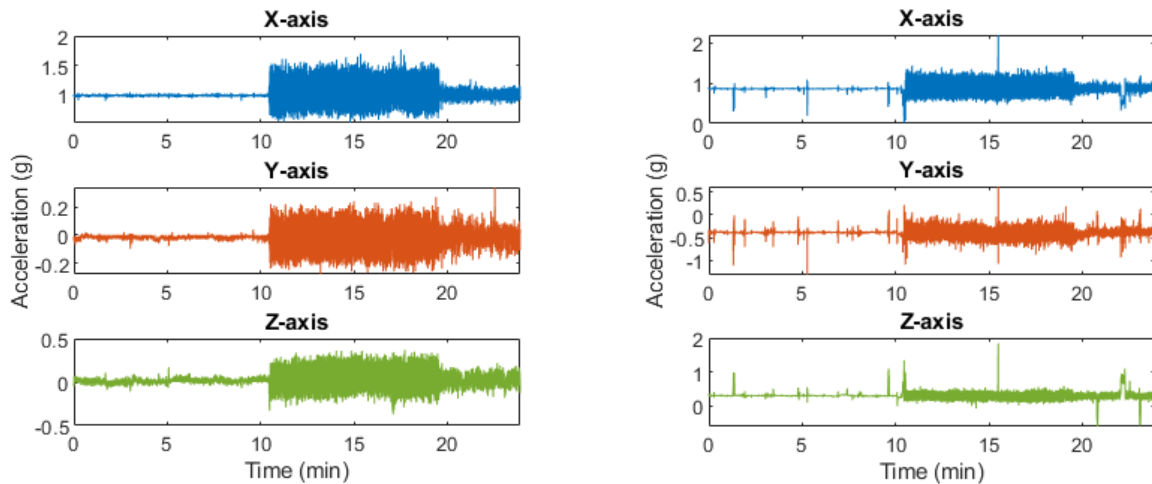
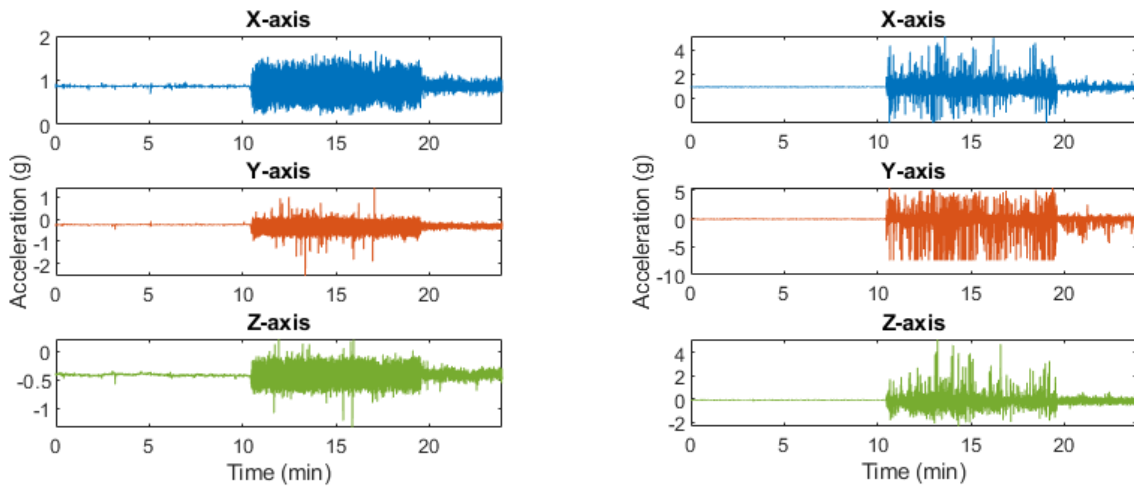


Figure 51: Raw 3D acceleration signals measured by the InertiaLab's IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 1.



(a) Chest

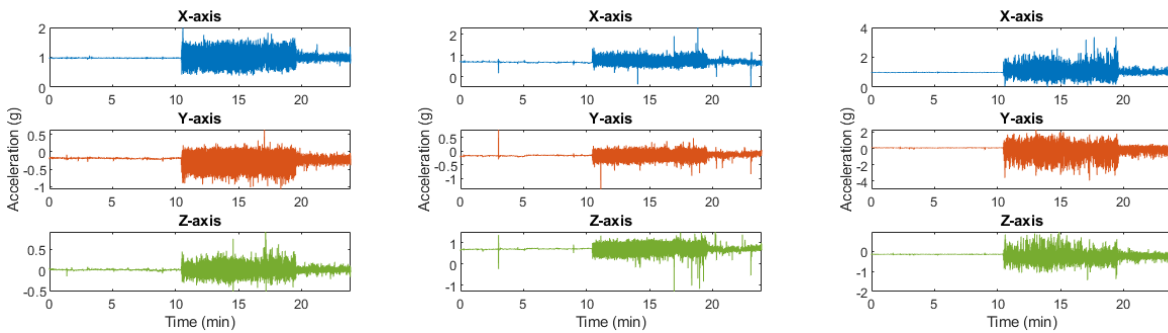
(b) Right wrist



(c) Left waist

(d) Right ankle

Figure 52: Raw 3D acceleration signals measured by the InertiaLab's IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 2 (Trial 1).



(a) Right waist

(b) Back waist

(c) Right knee

Figure 53: 3D Acceleration signals measured by the InertiaLab's IMUs that were not used to predict the MC (right waist, back waist, and right knee), for Participant 2 (Trial 1).

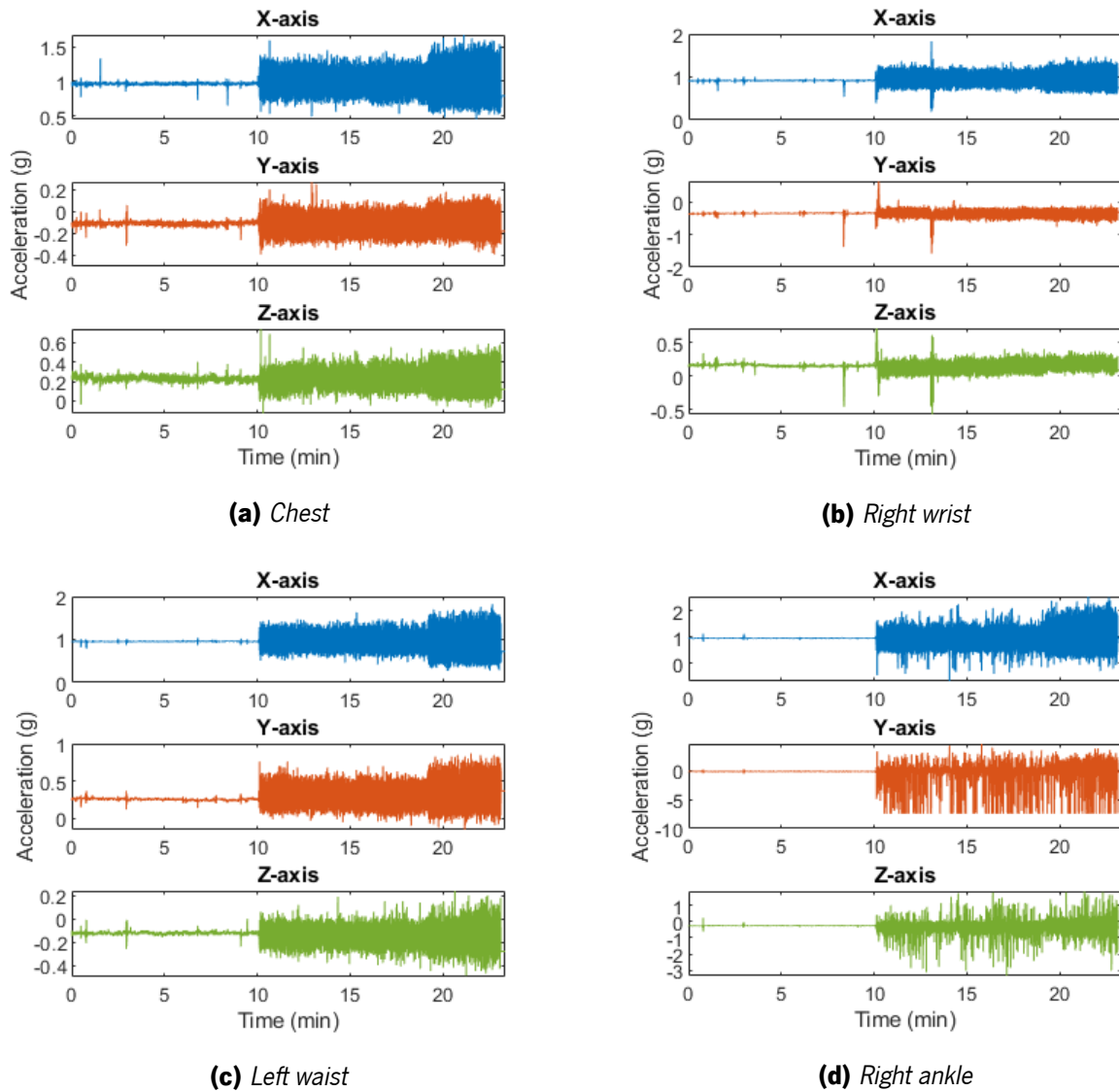


Figure 54: Raw 3D acceleration signals measured by the InertiaLab's IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 2 (Trial 2).

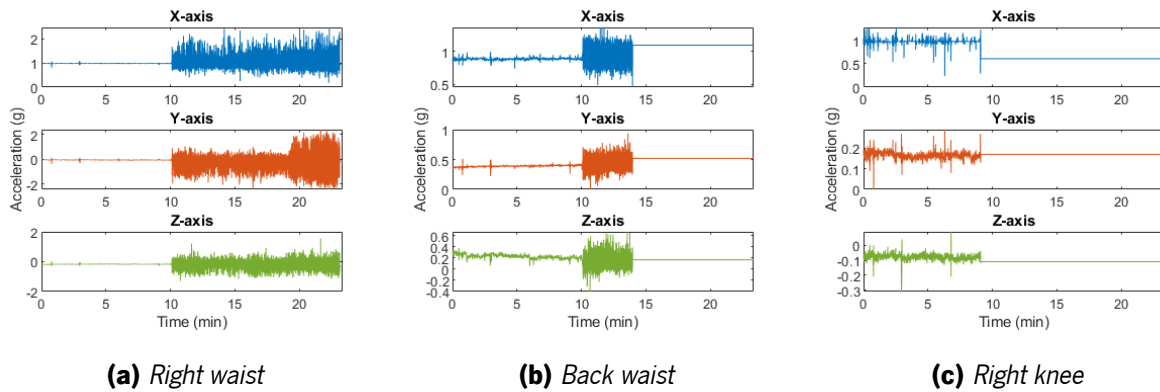
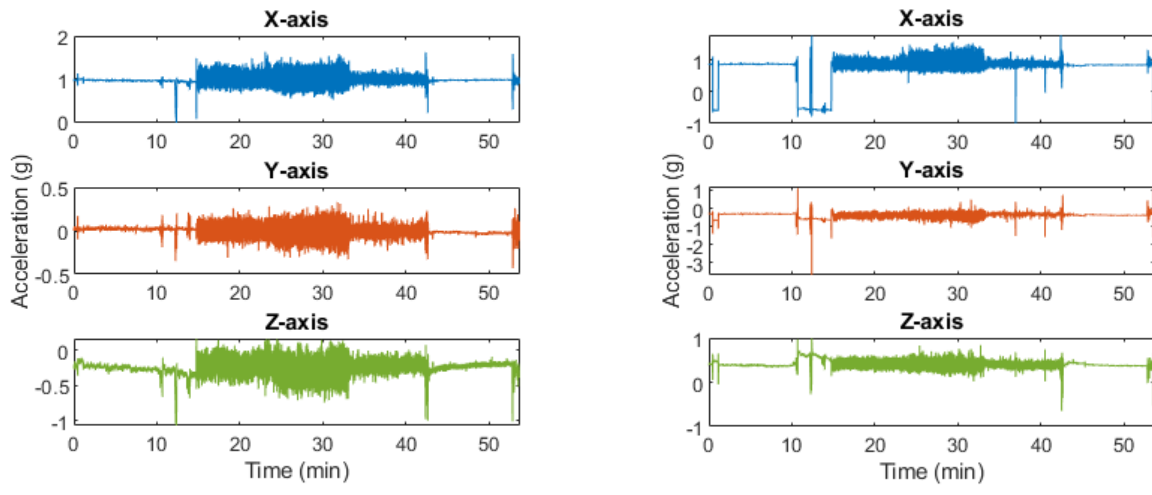
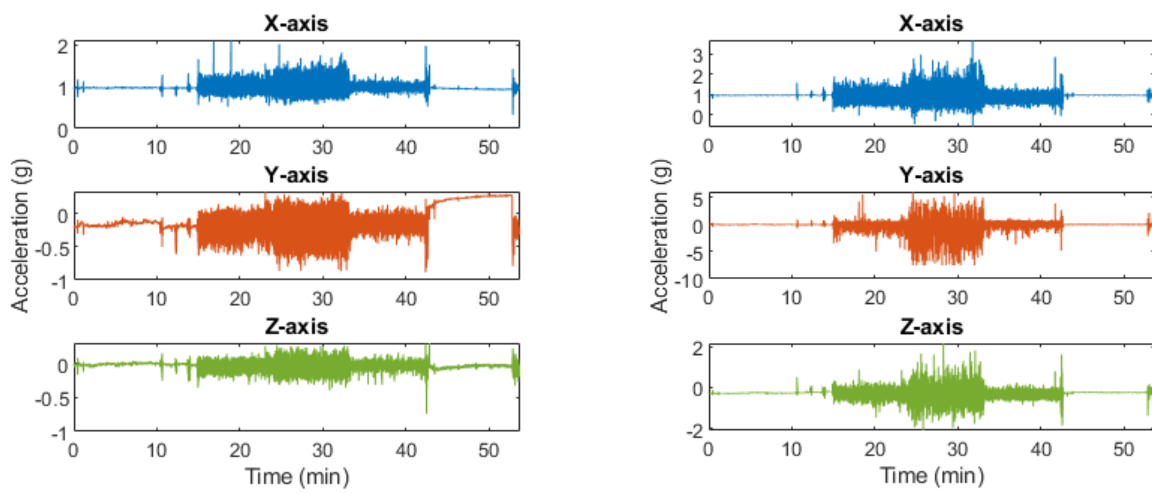


Figure 55: 3D Acceleration signals measured by the InertiaLab's IMUs that were not used to predict the MC (right waist, back waist, and right knee), for Participant 2 (Trial 2).

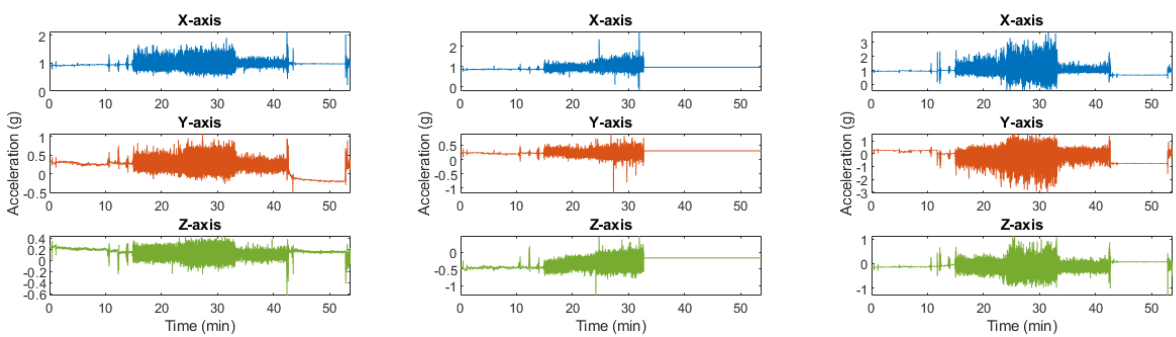


(a) Chest (b) Right wrist



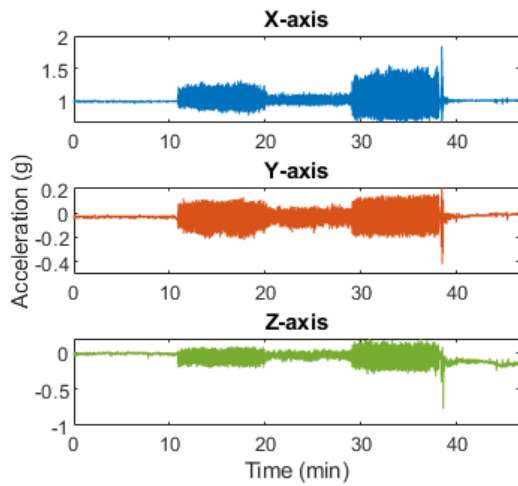
(c) Left waist (d) Right ankle

Figure 56: Raw 3D acceleration signals measured by the InertiaLab's IMUs that were used to predict the MC(chest, right wrist, left waist, and right ankle), for Participant 3.

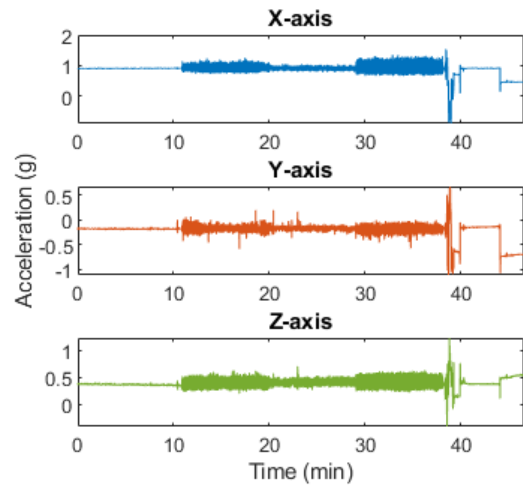


(a) Right waist (b) Back waist (c) Right knee

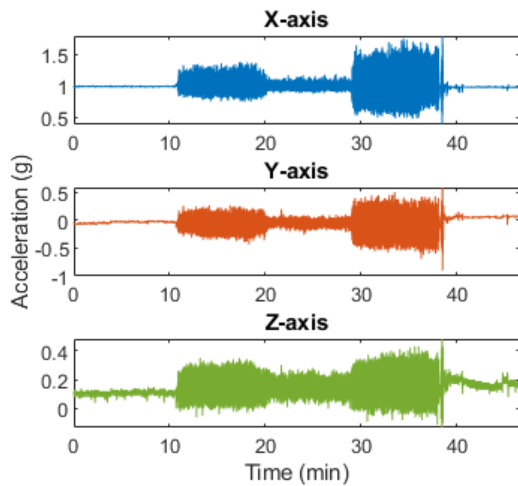
Figure 57: 3D Acceleration signals measured by the InertiaLab's IMUs that were not used to predict the MC (right waist, back waist, and right knee), for Participant 3.



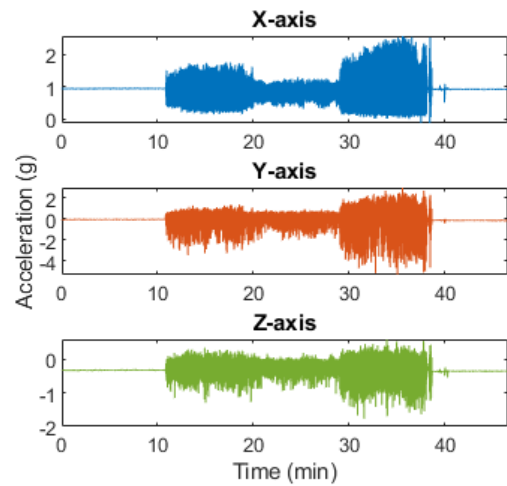
(a) Chest



(b) Right wrist

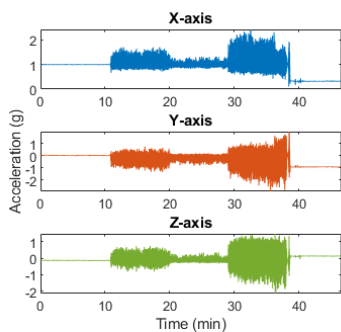


(c) Left waist

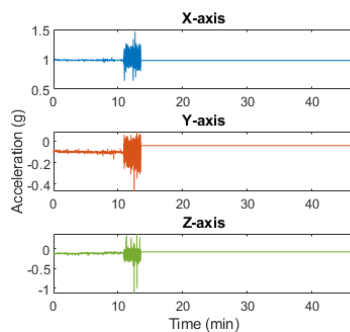


(d) Right ankle

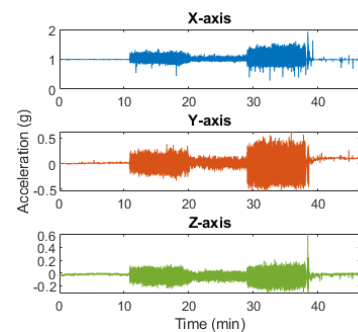
Figure 58: Raw 3D acceleration signals measured by the InertiaLab's IMUs that were used to predict the MC (chest, right wrist, left waist, and right ankle), for Participant 4.



(a) Right waist

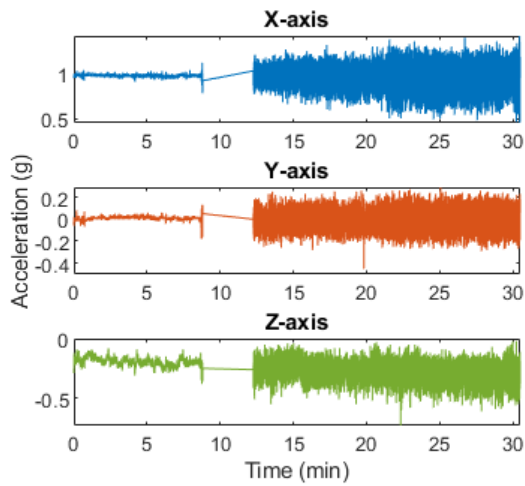


(b) Back waist

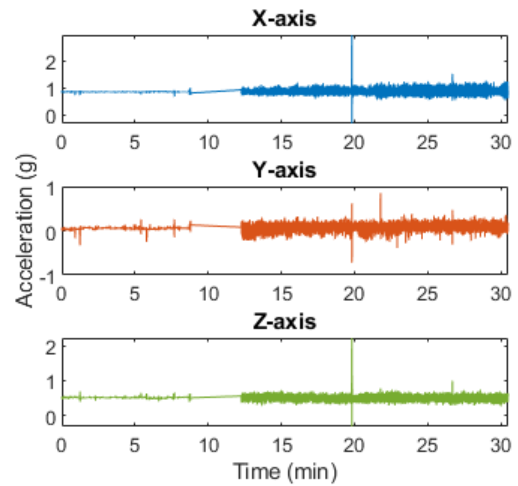


(c) Right knee

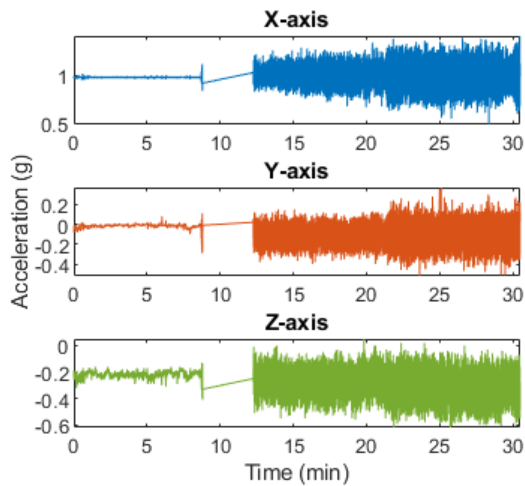
Figure 59: 3D Acceleration signals measured by the InertiaLab's IMUs that were not used to predict the MC (right waist, back waist, and right knee), for Participant 4.



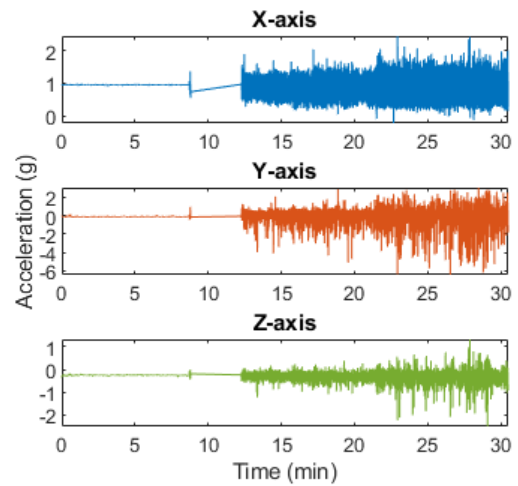
(a) Chest



(b) Right wrist

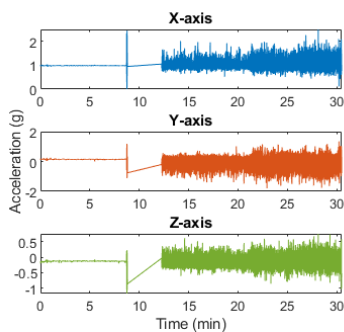


(c) Left wrist

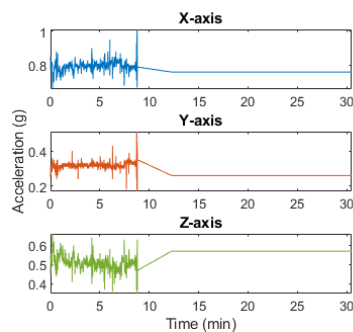


(d) Right ankle

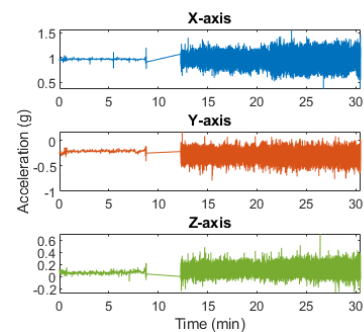
Figure 60: Raw 3D acceleration signals measured by the InertiaLab's IMUs that were used to predict the MC(chest, right wrist, left wrist, and right ankle), for Participant 5.



(a) Right wrist



(b) Back wrist



(c) Right knee

Figure 61: 3D Acceleration signals measured by the InertiaLab's IMUs that were not used to predict the MC (right wrist, back wrist, and right knee), for Participant 5.