EBRAINS Closed loop AI and robotics workflows
(D5.11 – SGA3)

An integration of in-silico simulation capabilities of lower body assistive robotics in the HBP Neurorobotics Platform, based on the CESPAR (Closed-loop Exoskeleton Simulation for Personalised Assistive Rehabilitation) project - see Clarification about subject matter.

Figure 1: Human musculoskeletal systems and robotic assistive device in-silico simulation environment

CESPAR adds a feature to the HBP Neurorobotics Platform, to perform experiments with a closed-loop, biologically plausible control architecture. It comprises a neuromuscular model of human gait with an attached assistive robotic device (An exoskeleton for lower limb from Autonomyo SARL). The aim is to study simulated neurorehabilitation of patients with lower limb motor control diseases.

Project Number:	945539	Project Title:	HBP SGA3

Document Title:	EBRAINS Closed loop AI and robotics workflows
Document Filename:	D5.11 (D100) SGA3 M39 SUBMIT 230621.docx
Deliverable Number:	SGA3 D5.11
Deliverable Type:	Report
Dissemination Level:	PU = Public
Planned Delivery Date:	SGA3 M39 / 30 Jun 2023
Actual Delivery Date:	SGA3 M39 / 21 Jun 2023 (resubmitted 05 Dec 2023)
Author(s):	Berat DENIZDURDURAN, AIS (P148)
Contributor(s):	Berat DENIZDURDURAN, AIS (P148), contributed to Sections 1, 2, 3, 4 & 5 Aliaa DIAB, AIS (P148), contributed to Section 1, 2, 3, 4 & 5 Carla NANNINI, AIS (P148), contributed to Section 4 Raphael GAIFFE, AIS (P148), contributed to Section 3 Amalric ORTLIEB, AUTONOMYO (P149), contributed to Sections 2 & 4
WP QC Review:	Anne ELFGEN, JUELICH (P20)
WP Leader / Deputy Leader Sign Off:	Yannis IOANNIDIS, ATHENA (P133)
T7.4 QC Review:	Martin TELEFONT, Guy WILLIS, EBRAINS (P1)
Description in GA:	Neurobotics Platform release with improved functionality, extended content, and updated inventory of related closed loop models, tools, and services prepared for, or released through the EBRAINS portal (see Clarification about subject matter).
Abstract:	Musculoskeletal control poses computational challenges due to the complexity of controlling a dynamic system with redundant muscle units. Computational models of human motor control systems aim to understand human mobility and its limitations, incorporating both neuroscientific and biomechanical perspectives. While there have been advancements in explaining principles of human mobility, the application of computational models to motor control-related diseases and traumas remains a challenge. This study proposes an optimisation and learning framework for addressing muscle control redundancies and controlling assistive robotics. Model meta-heuristic optimisation generates energy-efficient skeletal trajectories, while deep reinforcement learning determines stimuli for assistive robotics. Integration of the Human Brain Project's Neurorobotics Platform allows for realistic closed-loop experiments and the evaluation of biologically plausible motor control architectures, particularly in clinical neurorehabilitation. The use of a spinal cord model and the inclusion of exoskeletons permits the assessment of their impact of the patient's motor control. This research contributes to improving computational models and understanding motor control in individuals with motor control diseases. The use of large-scale simulations and the EBRAINS research infrastructure, specifically the Piz Daint supercomputer, supports the development and evaluation of these advanced functionalities.
Keywords:	Neurorobotics, simulation, closed-loop experiments, assistive robotics, deep neural networks, control architecture
Target Users/Readers:	Computational neuroscience community, assistive robotics community. computer scientists, Consortium members, AI and ML researchers, roboticists, general public, neuroscientific community, neurorehabilitation community, Platform users, Postdocs, PhDs and Master's students.

Table of Contents

Table of Figures

History of Changes made to this Deliverable (post Submission)

Date	Change Requested / Change Made / Other Action
June 21, 2023	Deliverable submitted to EC
Dec 1, 2023	Resubmission with specified changes requested in Review Report Main changes requested: 1. The deliverable forms the expectation, that the document should present the workflow, including the presentation of the parts, the connection of the parts, the motivation, of the selected solution, as well as a demonstration application', with the assumed goal to present the features and possible application of the EBRAINs robotics simulation. However, the submitted deliverable is not describing all these parts with sufficient details, nor is there a publication mentioned linked to example Figure 5, and the presented results that present the details sufficiently. 2. The deliverable, needs to improve the documentation of: 1) workflow, including the presentation of the parts, the connection of the parts, the motivation, of the selected solution; the demonstration application and more detailed analysis of the derived results; most importantly, the presentation of the workflow needs to include a justification of the chosen approach, to allow new and future user to assess the quality, versatility, and readiness of the EBRAIN parts. 3. In a use case the integration of exoskeleton and human model is described. Has there been any experimental work to test the performance of the errors in the simulations? Is there any objective data to evaluate it? 4. Reference is missing on the Covariance Matrix Adaptation Evolution Strategy. Please clarify which fitness function was used to characterize the fitting between the two controllers used. 5. It is not clear which validation scheme was used for tuning hyperparameters. 6. Qualitative characterization of the optimization outcome is shown. However, no numerical values of the fitness function convergence and the final error between the human gait and the simulated gait are given. These are important to understand the quality of the achieved simulation. 7. Moreover, it is not clear how many different subjects were used to characterize human locomotion and how the associated parameters are extracted. 8. A comparison with simpler optimization strategies is also important to evaluate its suitability in solving the given problem. 9. Although the introduction of the deliverable mentions a reinforcement learning approach, its content does not describe such an approach. Results should be given on the RL approach or its reference removed. 10. Although the work described demonstrates the core design principle of closed-loop action-perception simulation the references to a closed-loop system may lead to the interpretation that the achieved control is a closed-loop one, where sensing the environment modifies the control strategy adaptively. The deliverable should better describe how this last closed-loop strategy was implemented.
Dec 5, 2023	Revised draft sent by WP to PCO. Main changes made, with indication where each change was made: 1. Added explanation in page 20 and 21, 2. Added explanation in page 12, 13, 14 and 15, indicated by red, new Figure 4 and 5 are added 3. Added explanation in page 15 and 16, 4. Added reference in page 18, 19, 20 and 21, indicated by red, an equation describing the fitness function with explanations are added 5. Added explanation in page 19, 6. Added explanation in page 20, indicated by blue, new Figure 7 is added 7. Added explanation in page 18, 8. Added explanation in page 17, 9. Added explanation in page 22, 23 and 24, indicated by blue, with equation describing the reward function, new Figure 10 and 11 are added 10. Added explanation in page 19,
	Revised version resubmitted to EC by PCO via SyGMa

Clarification about subject matter

The title and the task description of this deliverable, D5.11, as defined in the HBP SGA3 Grant Agreement, mistakenly duplicated those of deliverable D5.9:

- **D5.9 Title:** EBRAINS Closed loop AI and robotics workflows
- **D5.9 Description:** Neurorobotics Platform release with improved functionality, extended content, and updated inventory of related closed-loop models, tools, and services prepared for, or released through the EBRAINS portal.
- **D5.11 Title:** EBRAINS Closed loop AI and robotics workflows
- **D5.11 Description:** Neurobotics Platform release with improved functionality, extended content, and updated inventory of related closed-loop models, tools, and services prepared for, or released through the EBRAINS portal.[1]

The intended focus of the deliverable D5.11 is defined in the description of the Task responsible for it, Task T5.20, in the same SGA3 Grant Agreement, which reads as follows:

Task T5.20 Personalised human musculoskeletal model and exoskeleton coupling

CEoI Wave 2: Engagement of Industry, SMEs and start-ups

This Task will enhance the capabilities of HBP SGA3 WPO5.2 objectives. It will be a direct usage of the HBP NRP in the industrial and clinical applications. An add-on tool will be provided that will allow physiotherapists and patients to avoid exhausting experiments as well as allow exoskeleton companies to design their products while having an ability to adjust their solutions with a closed-loop simulation environment. The integrated add-on tool aims at enabling users (e.g. an exoskeleton or prosthesis company, biomechanics researcher, computational neuroscientists, physiotherapists) to simulate the interactions between brain and spinal cord models, external devices and the human musculoskeletal model of a patient. The tool will embed the biomechanics, virtual muscles, deep neural networks for the control and adjustment of the exoskeletons/prosthesis. The implementation will be based on a modular control architecture that will allow users to adapt their devices to study the interest of therapy regimes, and will focus on lower-limb rehabilitation strategies. The partner (company) will evaluate the maturity level of the developing technology/tool and elaborate a full exploitation plan for that technology, to be updated every six months. The plan will demonstrate that the technology will be ready for scientific and/or industrial exploitation through EBRAINS, and make special emphasis on the IP strategy, market approach, and the steps taken to eventually obtain further public and/or private funding.

List of Outputs to which this Task contributes:

- OP5.41 (M15-M30): Automation of the modelling and personalised human musculoskeletal simulations along with exoskeleton coupling.
- OP5.42 (M21-M35): Optimisation pipeline of the human gait and rehabilitation scheme.
- OP5.43 (M24-M39): Clinical experiments and validation of the add-on tool for the patients with motor control disorders.

[1] By the time this error was detected, it was too late to correct it by modifying the Deliverable title and description via an amendment of the Grant Agreement.

1. Introduction

The fundamental role of the Central Nervous System (CNS) lies in its ability to perceive changes in the environment and execute appropriate motor skills. These motor skills encompass a wide range of actions that individuals perform to achieve their goals, driven by amusement, survival instincts, or curiosity. Remarkably, the CNS can acquire new motor skills and integrate them with existing ones, thereby reorganising the sequence of learned actions and enhancing motor control capabilities. Research has shown that these learned motor skills can persist throughout an individual's lifetime, indicating the long-term storage capacity of the CNS through a mechanism of retained plasticity. However, there is still active research interest in understanding the division of labour involved in acquiring new motor skills and executing integrated ones, as well as the role of perception and sensory integration across distributed motor areas.

Controlling the musculoskeletal system poses a computational challenge because it involves managing a complex, dynamic system consisting of multiple interconnected bodies. These bodies have the ability to move in various ways, which are quantified by degrees of freedom. Degrees of freedom represent the number of independent variables required to describe the system's motion fully. Additionally, the musculoskeletal system possesses redundant muscle units, meaning there are more muscles available than strictly necessary for accomplishing a particular movement or task. In simpler terms, the challenge lies in effectively coordinating and managing the movements of a dynamic, multi-part system with many possible ways of moving and more muscles than needed.

Specifically, controlling skeletal joints with antagonistic muscle pairs is an ill-posed nonlinear problem that requires robust methods. To address this scientific challenge, computational models of human motor control systems have been developed with the aim of uncovering the underlying structure of human mobility, its capabilities and limitations. These models take into account both neuroscientific and biomechanical perspectives, as the motor control system represents a closed action-perception loop within the CNS. Specifically, neuromechanical simulations have been used to create computational models that evaluate and validate physically accurate movements of a human musculoskeletal system. While there have been significant advancements in biologically plausible motor control models that explain principles of human mobility such as goal-directed behaviours, walking, and running, the application of neuromechanical simulations to motor control-related diseases and traumas remains a significant challenge. Conditions like amputation, paraplegia, muscle weakness due to ageing, cerebral palsy and stroke pose complex problems that necessitate further exploration and development of accurate computational models to better understand and address them.

In this deliverable, we describe a two-fold optimisation and learning framework, the CESPAR exoskeleton simulator, that tackles the computational challenges arising from mechanically coupled assistive robotic control and human musculoskeletal system. This coupling allows us to study how the exoskeleton influences and interacts with the human musculoskeletal system. Essentially, we also explore how assistive robotics and the human body work together and influence each other in our computational simulations. In the first part of our framework, we employed meta-heuristic model optimisation to generate energy-efficient skeletal trajectories that closely mimic human movements. This optimisation process enabled us to obtain desired trajectories for the musculoskeletal system. In the second part, we utilised deep reinforcement learning techniques to determine a sequence of stimuli that should be applied to the actuators of the assistive robotics device to achieve the desired skeletal trajectories of an impaired movement. It is noteworthy that efficient construction of the desired muscle stimulus relied on integrating the state and control input in a closed-loop setting, emulating the proprioceptive integration observed in spinal cord circuits.

The Human Brain Project's Neurorobotics Platform[2] (HBP-NRP) aims to bridge the gap between computational models and physical reality by establishing connections between physical sensors, actuators, devices and simulated brains. The primary objective is to facilitate realistic closed-loop experiments that encompass the perception-cognition-action loop. By leveraging the capabilities of the HBP-NRP, computational models can be enhanced to gain a deeper understanding of the human

[2] https://neurorobotics.net/

motor control system. In this deliverable, we applied this approach to the field of clinical neurorehabilitation, specifically focusing on patients with motor control diseases.

Our work involved evaluating a biologically plausible motor control architecture by integrating exoskeletons/exosuits into the HBP-NRP. This integration allowed for the assessment of how these devices interact with the patient's motor control system. To accomplish this, we introduced an additional tool within the HBP-NRP that enables users, such as exoskeleton companies, biomechanics researchers and computational neuroscientists, to incorporate their models or devices into the human musculoskeletal model of the patient. This tool also facilitated the coupling of exoskeletons with the patient's musculoskeletal system, driven by a spinal cord model.

By leveraging the HBP-NRP and utilising this novel tool, we anticipate significant advancements in the field of clinical neurorehabilitation. The ability to integrate exoskeletons and assess their impact on the patient's motor control system within a realistic closed-loop experiment will provide valuable insights for designing effective rehabilitation strategies. This research endeavour represents a crucial step toward improving the quality of computational models and expanding our understanding of motor control in individuals with motor control diseases.

This deliverable addresses two key aspects of our software development activities, specifically highlighting their implications for users in the fields of neurorehabilitation and assistive robotics. The first aspect is a tool that allows integration of assistive robotic devices, scaling them to a patient's anatomy and coupling them to the patient's musculoskeletal model, without limiting the joints' range of motion. The second is an optimisation and learning pipeline that finds optimal musculoskeletal control actions to reproduce a human movement in simulation and also allows for the design and integration of control architecture for assistive robotic devices. To achieve the second objective, we employed large-scale simulations utilising the EBRAINS infrastructure, predominantly leveraging the Piz Daint supercomputer located in Lugano, Switzerland.

By integrating scaling techniques into our software, we aimed to enhance the applicability and versatility of the developed solutions for users in the neurorehabilitation and assistive robotics industries. The automated adjustment of human musculoskeletal modelling provides a flexible framework that can be tailored to individual users' needs, enabling personalised and optimised rehabilitation or assistance. Furthermore, coupling patients' musculoskeletal models with assistive robotic devices allows for a better understanding of human-machine interactions and makes it possible for more fine-grained assistive control schemes to be implemented.

Our optimisation and learning pipeline should play a crucial role in achieving human-like musculoskeletal movement and controlling the assistive robotic devices. By mimicking human movements, we showed that we could generate realistic trajectories that should facilitate effective rehabilitation or assistance. Simultaneously, the control architecture of the assistive robotic devices should benefit from a robust learning process that adapts to the user's specific needs and requirements, resulting in more efficient and intuitive device operation.

To support the development and evaluation of these advanced functionalities, we have leveraged the powerful computational capabilities of the EBRAINS infrastructure. The Piz Daint supercomputer, known for its high performance, has enabled us to conduct large-scale simulations that validate the effectiveness and feasibility of our proposed approaches. By utilising these resources, we can optimise our software and ensure its practical viability before implementation in real-world scenarios.

In summary, this deliverable describes the integration of scaling, automated adjustment of human musculoskeletal modelling and coupling with assistive robotic devices, all within an optimisation and learning pipeline. Leveraging the computational power of the EBRAINS infrastructure, specifically the Piz Daint supercomputer, we have demonstrated the potential value of our software in the context of neurorehabilitation and assistive robotics. This research should pave the way for improved solutions that can significantly benefit users in these domains, help people who have lost the ability to walk, to do so in a natural manner with the assistance of an exoskeleton and enabling more efficient and intuitive control of assistive robotic devices.

2. Neuromechanical models of human locomotion

Walking is a critical aspect of daily life, and the ability to move without pain, fatigue or gait deviation is strongly associated with a good quality of life. Moreover, mobility plays a crucial role in maintaining overall health, as it provides protective effects against various conditions such as heart disease, osteoporosis, diabetes and dementia (CDC of the USA, 2023[3]). However, neurological disorders like stroke often lead to significant motor impairments, including walking difficulties. Stroke survivors typically undergo rehabilitation to enhance their sensory, motor and functional capabilities. Robotic solutions can aid in this rehabilitation process, particularly during the acute phase, by increasing the intensity of therapy, and may also be useful for aiding post-surgery rehabilitation, in cases where patients need help in "relearning" how to use their legs. The integration of robotics, such as the use of exoskeletons for gait rehabilitation, continues to improve and influence rehabilitation practices. However, there are several unresolved challenges that impede the rapid adoption of exoskeletons for stroke patients.

One of the challenges lies in developing an effective interaction strategy between the exoskeleton and the human user. This requires ensuring efficient back-drivability of the device and considering the user's intention in the control process. Currently, many existing devices primarily target individuals with complete paraplegia, where pre-computed gait trajectories can be implemented without much adaptation. However, a distinction must be made between mobilisation and walk-assistance. Mobilisation involves providing a defined desired gait using precomputed trajectories, while walk-assistance requires continuous adaptation of joint actuation to the user's intent and capabilities. To design personalised exoskeleton control and rehabilitation schemes, it is crucial to quantify the patient's intent, capabilities and restrictions. This involves adapting precomputed gait trajectories to the patient's musculoskeletal conditions, considering the capabilities of the exoskeleton, and optimising the rehabilitation protocol accordingly.

To address these challenges, this project established a closed-loop musculoskeletal and exoskeleton simulation, allowing clinicians to study rehabilitation strategies and enabling exoskeleton companies to customise their physical devices to meet the specific needs of individual patients. The integration of this simulation within the HBP-NRP provides a virtual environment for clinicians and researchers to explore and refine rehabilitation approaches. By leveraging the capabilities of the HBP-NRP, valuable insights can be gained, contributing to the advancement of personalised rehabilitation and the development of more effective exoskeleton systems.

2.1 Early Models of Locomotion

Extensive research spanning several decades has been dedicated to the investigation of locomotion control, particularly in the context of animal locomotion. Early studies involved surgical removal of the cerebrum and transection of the spinal cord, allowing researchers to examine locomotion in the absence of descending signals from the brain and spinal cord above the transaction point (Flourens, 1824[4]; Sherrington, 1910[5]). Notably, Sherrington observed that the release of a hind limb from a flexed position triggered a series of locomotor-like movements characterised by alternating contractions of the muscles. This observation led to the proposition that locomotion control relies on a chain of spinal reflexes. However, further investigations challenged this reflex-based perspective.

[3] Physical activity helps prevent chronic diseases (2023) Centers for Disease Control and Prevention. Available at: https://www.cdc.gov/chronicdisease/resources/infographic/physical-activity.htm#:~:text=Regular%20physical%20activity%20helps%20improve,depression%20and%20anxiety%2C%20and%20dementia.

[4] Flourens, MJP. (1824). 'Recherches expérimentales sur les propriétés et les fonctions du système nerveux dans les animaux vertébrés', Paris: Crevot.

[5] Sherrington, C.S. (1910) 'Flexion-reflex of the limb, crossed extension-reflex, and reflex stepping and standing', The Journal of Physiology, 40(1-2), pp. 28-121. doi:10.1113/jphysiol.1910.sp001362.

T.G. Brown, a student of Sherrington, conducted experiments with spinally transected cats and questioned the reflex-driven nature of locomotion control. Despite the transection of sensory nerves entering the spinal cord, Brown observed coordinated contractions of the extensor and flexor muscles in the hind limbs (Brown, 1911[6]). This finding suggested the presence of an intrinsic factor within the spinal cord capable of generating alternating locomotor-like motions without the involvement of descending signals from the brain or sensory inputs. This intrinsic factor was later referred to as the Central Pattern Generator (CPG) by Grillner (Grillner, 1975[7]). CPGs have since been identified in various species of vertebrates and invertebrates, highlighting their widespread presence and significance in locomotion control (Grillner, 1975).

The discovery and characterisation of CPGs have greatly advanced our understanding of the underlying mechanisms of locomotion. These intrinsic neural networks within the spinal cord play a vital role in generating coordinated patterns of muscle activity necessary for locomotor behaviours. Their existence across different species suggests a fundamental, conserved role in the control of movement. The study of CPGs continues to shed light on the intricate interplay between neural circuits and motor behaviours, furthering our knowledge of locomotion control in both physiological and pathological conditions.

2.2 Feedback Models

While CPGs have been extensively studied as a fundamental mechanism underlying locomotion control, there are research groups that explore locomotion control models without relying on a CPG layer. In these alternative models, locomotion is predominantly driven by reflexive feedback control mechanisms that utilise sensory inputs from various muscles or muscle groups, adapting to the environment and the body's requirements (Geyer & Herr, 2010[8]; Song & Geyer, 2015[9]).

Unlike CPG-based control, which involves intrinsic neural networks within the spinal cord, reflexive feedback control relies on sensors distributed throughout the body, including the spinal cord and, in the case of humans, the supraspinal nervous system. These sensors provide essential feedback signals that inform the control system about the current state of the muscles and the environment, allowing for adaptive and context-dependent locomotion control (Geyer & Herr, 2010; Song & Geyer, 2015).

By focusing on reflexive feedback control, these research efforts contribute to a comprehensive understanding of the different mechanisms involved in locomotion control. While CPGs provide an important framework for generating rhythmic patterns of movement, reflexive feedback control highlights the significance of sensory feedback and the integration of multiple sources of information to achieve effective locomotion in diverse conditions. The investigation of these alternative control strategies offers valuable insights into the complexity and versatility of locomotion control mechanisms and their role in adapting to changing environments.

2.2.1 H. Geyer's Reflex Controller

In a study conducted by H. Geyer and colleagues in 2010, a bipedal walking model was introduced that incorporated muscle reflexes to emulate the intricate interactions between the environment and the body during walking (Geyer & Herr, 2010). The model aimed to capture the fundamental

[6] Brown, T.G. (1911) 'The intrinsic factors in the act of progression in the mammal', Proceedings of the Royal Society of London. Series B, Containing Papers of a Biological Character, 84(572), pp. 308–319. doi:10.1098/rspb.1911.0077.

[7] Grillner, S. (1975) 'Locomotion in vertebrates: Central mechanisms and reflex interaction', Physiological Reviews, 55(2), pp. 247–304. doi:10.1152/physrev.1975.55.2.247.

[8] Geyer, H. and Herr, H. (2010) 'A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities', IEEE Transactions on Neural Systems and Rehabilitation Engineering, 18(3), pp. 263–273. doi:10.1109/tnsre.2010.2047592.

[9] Song, S. and Geyer, H. (2015) 'A neural circuitry that emphasizes spinal feedback generates diverse behaviours of human locomotion', The Journal of Physiology, 593(16), pp. 3493–3511. doi:10.1113/jp270228.

principles of legged mechanics and replicate a stable and robust walking gait similar to that observed in humans.

The basis of this model lay in a conceptual bipedal spring-mass framework, which served as a foundation for understanding the dynamics of bipedal locomotion. By augmenting this framework with principles derived from legged mechanics, the model integrated crucial aspects such as compliant leg behaviour, ground reaction forces, and energy transfer mechanisms during walking (Geyer & Herr, 2010).

The inclusion of muscle reflexes in the control scheme added an additional layer of realism to the model. These reflexes, driven by sensory feedback from the environment and the body, enabled the adaptation of the walking pattern in response to changes in terrain or disturbances. By incorporating reflexive control mechanisms, the model demonstrated the ability to generate walking gaits that exhibit stability and robustness similar to those observed in human locomotion (Geyer & Herr, 2010).

The work by Geyer and colleagues highlighted the significance of legged mechanics and muscle reflexes in achieving realistic bipedal walking. By encoding these principles into their model, a comprehensive understanding of the underlying mechanisms and control strategies involved in human walking could be gained. Such models serve as valuable tools for studying locomotion and contribute to the development of bipedal robots and assistive devices that aim to replicate human-like walking patterns and enhance overall stability and adaptability.

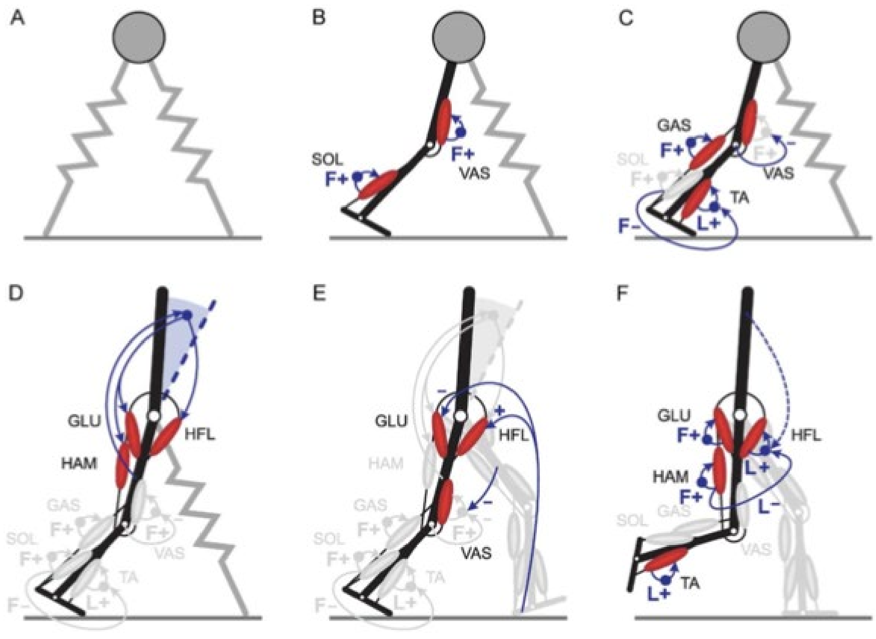

Figure 2: Geyer Model Architecture [Geyer & Herr, 2010]

As depicted in Figure 2, the model constructed by Geyer and colleagues in 2010 represents a simplified human body configuration, consisting of a trunk and two legs composed of hip, knee and ankle joints. Figure 2A depicts the starting point of the lower body model as a point-mass trunk with two massless springs representing the legs. In Figure 2B, positive force feedback (F+) was added to the SOL and VAS muscles to generate compliant leg behaviour. In Figure 2C, a positive force feedback was added to the GAS muscle to prevent knee overextension and inhibition of the VAS muscle. In Figure 2D, the trunk was compelled to a lean angle by the hip muscles (GLU, HAM and HFL), with the HAM muscle preventing hyperextension of the knees. In E, swing motion of the legs was aided by increasing/decreasing the constant stimulation of HFL and GLU, respectively, and VAS was inhibited proportionally to the load that the opposite leg bears. Finally, in Figure 2E, the model is complete with the addition of negative length feedback (L-) from the HAM muscle, positive length feedback (l+) from the TA muscle to flex the ankle, and positive force feedback from GLU and HAM muscles enabling the leg to be retracted and straightened during the swing. The leg segments are divided into three segments each, enabling a direct comparison to the corresponding joints found in human locomotion. In terms of muscle actuation, each sagittal leg incorporates seven Hill-type muscles, mirroring the muscle groups utilised by humans during walking.

The 2D neuromuscular model developed by Geyer *et al*. predominantly operates through a feedback mechanism, devoid of a feedforward component. This characteristic highlights the reliance on sensory information and reflexive responses to generate a stable, human-like walking gait. The absence of a feedforward component signifies that the model does not incorporate anticipatory control strategies.

Expanding upon Geyer *et al*.'s work leads to introducing a 3D neuromuscular model. This enhanced model builds upon the principles established by Geyer *et al*., aiming to achieve a more comprehensive representation of human locomotion, within a three-dimensional context. These models, through their inclusion of realistic joint structures and muscle actuation patterns, contribute to the understanding of biomechanics and provide valuable insights for the development of humanoid robots and assistive technologies aimed at achieving human-like locomotion.

2.2.2 Adapted musculoskeletal model in CESPAR

In the CESPAR project, we made adaptations to the model proposed by Geyer *et al*. (2010) to facilitate various experimental investigations. Our modifications entailed simplifying the model and shifting the control paradigm to a 2D framework, thereby eliminating the inclusion of the 3D HAB (hip abductor) and HAD (hip adductor) muscles present in the original model as shown in Figure 3 (left).

By making these adjustments, we aimed to streamline the experimental setup and focus on specific aspects of locomotion control within a 2D domain. This allowed for a more targeted analysis of the underlying mechanisms and dynamics involved in generating stable walking gaits. By removing the 3D HAB and HAD muscles, we effectively reduced the complexity of the model, while maintaining its core features and functionality.

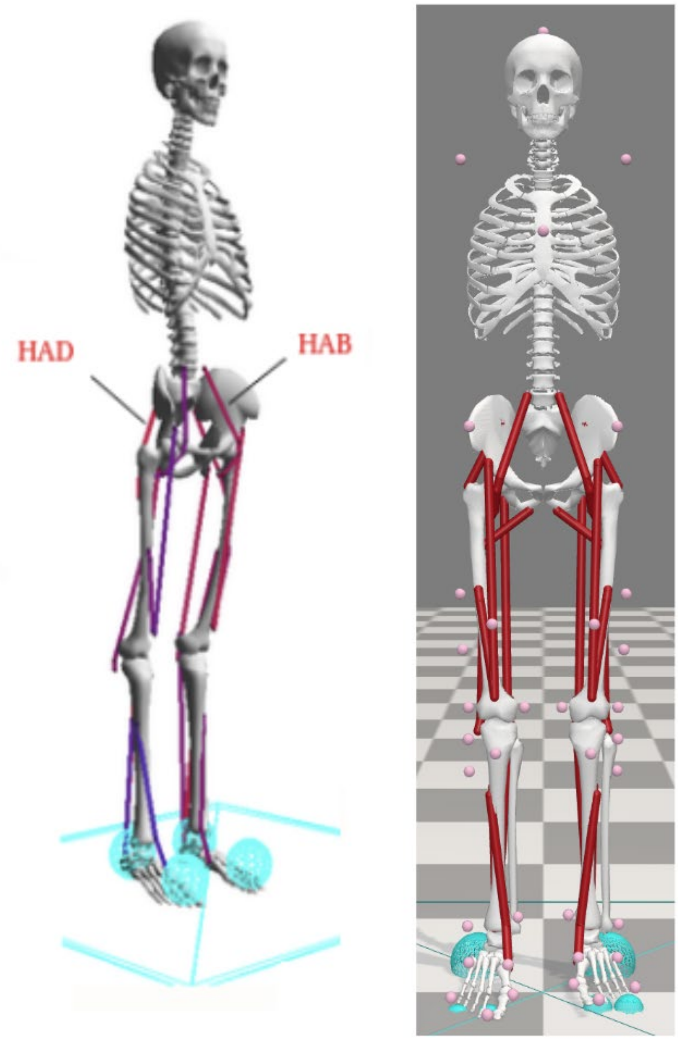

Figure 3: (left) Removed muscles. (right) Adapted Musculoskeletal model in CESPAR.

As a result, the revised model now incorporates a total of nine muscles per leg, namely GAS (gastrocnemius), BFSH (biceps femoris short head), GLU (gluteus maximus), HAM (hamstrings), HFL (hip flexors), RF (rectus femoris), SOL (soleus), VAS (vastus), and TA (tibialis anterior). In addition, the model comprises seven segments, encompassing the thigh, shank, foot, and trunk representing the upper body as shown in Figure 3 (right). Within this framework, three internal degrees of freedom (DOFs) are taken into account, specifically hip flexion, knee, and ankle DOFs, resulting in a total of six internal DOFs in the musculoskeletal model employed for this project. Furthermore, the model retains the inclusion of the four contact points per foot.

The utilised model is characterised by a state vector consisting of 85 variables, including: muscle states (length, velocity, and force) for both legs (54 variables); joint states and angles (16 variables) of the hip, knee and ankle; ground contact information (6 variables), reflecting ground reaction forces (GRF); and pelvis state (height, pitch, roll and 6 velocities), totalling 9 variables. The outputs of the controller are described by an 18-dimensional vector representing the muscle activations of the 9 muscles in each leg, with activation values ranging between 0 and 1. These activations are generated by the spinal reflex modules, contributing to the control of motion in the model.

In summary, this model connects the three segments of each leg through joints and governs its motion through the excitation of muscles. It offers a comprehensive representation of the musculoskeletal system and provides a detailed set of variables and activations that capture the dynamics and control of locomotion.

2.3 Assistive robotic device coupling with human musculoskeletal system

The primary challenge associated with fine-tuning exoskeleton controllers in the absence of a closed-loop simulation lies in the substantial time and effort required from physiotherapists and patients to execute this task. Conducting tests with human subjects under the supervision of physiotherapists proves to be not only time-intensive and expensive but also an unpleasant experience for both patients and practitioners. The core objective of the presented deliverable is to automate the personalization process of exoskeleton controllers and rehabilitation schemes, utilizing the closed-loop simulation architecture of the HBP-NRP along with biologically plausible models and controllers. The tool for personalized exoskeleton adjustment and rehabilitation suggestions aligns with the principles of HBP-NRP, enabling researchers to replicate traditional experiments in a virtual environment before applying their hypotheses to real-world scenarios. The deliverable involves the utilisation of an existing exoskeleton, Autonomyo, previously tested for Spinal Cord Injury (SCI) patients and developed by the Laboratory of Robotic Systems (LSRO) at EPFL Switzerland. Autonomyo, a walk assist exoskeleton with six actuated degrees of freedom, addresses key points such as the abduction/adduction of the hip joints in addition to the flexion/extension of the hip and knee joints to enhance lateral balance. The control over hip adduction/abduction is crucial for dynamic control of lateral step length, a significant factor in stability during double support phases. Autonomyo, weighing less than 25 kg, incorporates cable-driven transmissions from electrical motors, all positioned at the back, contributing to its highly backdrivable nature.

This deliverable was dedicated to addressing the challenges associated with predictive neuromechanical simulations and neurorehabilitation. The primary objective was to develop predictive simulations for exoskeleton applications within the context of neurorehabilitation. The overarching idea is to generate motions and desired behaviours without direct reliance on experiments involving human subjects and rehabilitation assistants. The focus of these studies is on comprehending the impact of assistive robotics devices on human subjects. By successfully predicting motions that align with the needs of human subjects using exoskeletons, the aim is to expedite rehabilitation treatments while minimising the reliance on experiments involving patients and rehabilitation assistants. This approach not only accelerates patient rehabilitation but also reduces the associated burden on both patients and rehabilitation assistants, thereby potentially lowering the overall procedural costs. Introducing such simulation platforms aims at a potential enhancement to streamline these experiments and facilitate necessary adjustments of the exoskeleton according to the specific requirements of the subjects involved.

As a part of the deliverable, we integrated the Autonomyo exoskeleton into the OpenSim musculoskeletal simulation framework which is an open-source software to facilitate biomechanical modeling, simulation and analysis (see Figure 4[10]), aiming to establish a robust mechanical coupling between the exoskeleton and the human musculoskeletal model. With this integration, we enhanced the capabilities of OpenSim to delve into the interconnected mechanics of the human musculoskeletal system and exoskeletons. Our project, CESPAR, extends the functionalities of OpenSim to specifically focus on the study of the interaction and integration between exoskeletons and the human body. This integration offers the first proof-of-concept enabling realistic simulations and evaluations of exoskeleton-assisted human locomotion, and represents a significant enhancement of the capabilities of the HBP Neurorobotics Platform, not to mention an advance in neurorobotics as a discipline. To ensure an accurate representation of the exoskeleton, meticulous adjustments and validations were conducted on the inertial properties of its various components. Through comprehensive analysis and experimental validation, the exoskeleton model's inertial parameters were refined, contributing to the fidelity and realism of the simulation.

Building upon the validated exoskeleton model, subsequent modifications and scaling were applied to align it precisely with the human musculoskeletal model, primarily focusing on achieving compatibility in the sagittal plane. By adjusting the exoskeleton's dimensions, joint positions and

[10] The details of this integration are provided as part of the source code documentation at https://github.com/alpineintuition/cespar/blob/main/md_files/opensim.md.

segment lengths, an initial configuration was established that closely resembled the human musculoskeletal system's biomechanical characteristics. To achieve a physically integrated coupling between the exoskeleton and the human musculoskeletal model, straps were employed to attach the exoskeleton securely to the corresponding anatomical landmarks. This mechanical connection ensured a reliable and stable interaction between the two systems during simulations and provided a foundation for analysing the dynamic interactions between the exoskeleton and the underlying musculoskeletal structure. The development of the automation of the alignment and scaling between human musculoskeletal system and exoskeleton was performed jointly by Alpine Intuition and Autonomyo. Autonomyo also provided engineering support for building the model generator and automatic scaling scripts that apply the individualised scaling factors to each of the following five body parts: torso, pelvis, femurs, tibias and feet, along with their corresponding joints in the exoskeleton.

The final implementation achieved congruence in terms of size and geometry between the musculoskeletal and exoskeleton models. This alignment significantly reduced assembly errors and joint range limitations, enhancing the overall accuracy of the simulation. The primary objective was to establish a seamless, synchronised movement by both the exoskeleton and the musculoskeletal system. By enabling the exoskeleton to faithfully replicate the movements generated by the musculoskeletal system, and vice versa, a fully integrated and mechanically optimised coupling was achieved. This coupling technique paves the way for more efficient and natural human-assistive robotic devices interactions.

To ensure the stability and consistency of the coupling in a simulation environment, point-in-line constraints were employed. These constraints acted as virtual anchors, allowing precise control of the interaction forces and ensuring a stable, reliable mechanical connection between the exoskeleton and the musculoskeletal model. To foster collaboration and facilitate further advancements, the initial implementation of this coupling methodology and the developed simulation framework, were shared with the HBP Neurorobotics team. This collaborative effort aimed to enhance the HBP Neurorobotics Platform, integrating our new software into its framework. By doing so, researchers and practitioners in the field would gain access to a powerful tool for studying and optimising exoskeleton-assisted locomotion, ultimately enhancing the development and deployment of exoskeleton technologies in real-world applications.

The procedure of the mechanical coupling in simulation between exoskeletons and human musculoskeletal system is the following:

Adding Exoskeleton model to the musculoskeletal model

To incorporate the exoskeleton into the musculoskeletal model, one has to include the exoskeleton components into .osim file of the target musculoskeletal model. Then it is necessary to establish the connections between the exoskeleton parts by defining their relationships. In the <parent_body> subsection, specify the part to which the exoskeleton component will be attached. It is recommended to have one fixed exoskeleton part, serving as an anchor, and then connect the remaining exo parts. For instance, in the context of a lower limb exoskeleton, the exo torso is initially fused to the musculoskeletal pelvis. The attachment sequence for exo parts (e.g., exo's feet ⟶ exo's shins ⟶ exo's femur ⟶ exo's hips ⟶ exo's torso) is preferable over connecting them with their musculoskeletal counterparts to prevent misalignment. To link exo parts, create a joint using options like WeldJoint, PinJoint, SliderJoint, BallJoint, EllipsoidJoint, FreeJoint, or CustomJoint. For instance, using the PinJoint for foot joint, specify the parent body knee joint to which it is attached and define the joint position in both parent and child body frames through <location_in_parent> and <orientation_in_parent>. After adding the body part and creating the joint, use the Opensim window's navigator tab, as in Figure 4, to edit the parent and child frames, aligning them as desired.

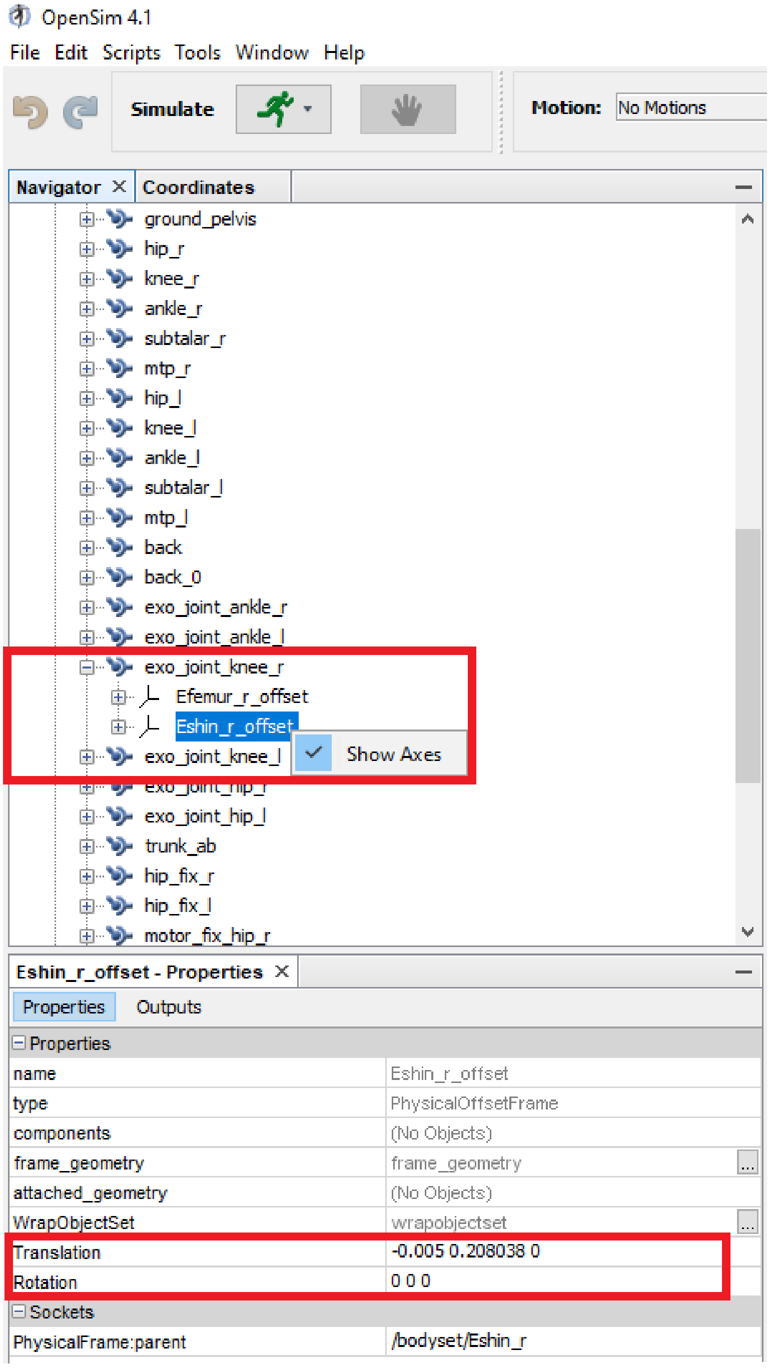

Figure 4: Coupled system's translation and position details in OpenSim

In OpenSim 4.1, it is necessary to input the exact mass value in the 'Mass' field of the post-scaling model as what is initially specified in the current model's 'Mass' field. This is because the scaling algorithm multiplies each body part's mass by the product of the part's scaled factors and then multiplies it by the ratio of the new target mass to the original mass. Consequently, regardless of whether the scaled model is larger or smaller, the mass value entered should match that of the original model. This approach ensures that the scaled parts have their mass adjusted in relation to their scaling factor, resulting in a total mass for the accurate new model. This practice eliminates the need for manually calculating the new mass of the scaled model. Concerning inertias, the OpenSim algorithm considers only the total scaled mass to total original mass ratio, not the scaled factors. Therefore, the required mass value for inertia calculations is the original mass multiplied by the scaling factor of a body part. Consequently, to obtain the correct inertia tensor for each scaled body part, the scaling process needs to be repeated for each distinct scaled body part. This procedural aspect proves useful for aligning the rotation axis of exoskeleton joints with their musculoskeletal counterparts or adjusting the exoskeleton width to match that of the musculoskeletal model, as observed in the discrepancy between the skeleton's feet and the exoskeleton's feet.

Integrating Actuators

In this section, incorporate the available actuators for your exoskeleton, ensuring attention to proper indentation. The Figure 5 below provides a visual representation of the actuators' arrangement in the .osim files, with these actuators corresponding to the exo joints equipped with motors.

```
            <CoordinateActuator name="exo_hip_motor_flex_left">
                <!--Flag indicating whether the force is disabled or not. Disabled means that the force is not active
                <isDisabled>false</isDisabled>
                <!--Minimum allowed value for control signal. Used primarily when solving for control values.-->
                <min_control>-Inf</min_control>
                <!--Maximum allowed value for control signal. Used primarily when solving for control values.-->
                <max_control>Inf</max_control>
                <!--Name of the generalized coordinate to which the actuator applies.-->
                <coordinate>exo_hip_l</coordinate>
            </CoordinateActuator>
        </objects>
        <groups />
    </ForceSet>
```

Figure 5: Inserting actuations into the Exoskeleton joints

Scaling Tool

To align the selected exoskeleton with the musculoskeletal model, the Scaling Tool can be employed. As its name implies, this tool facilitates the adjustment of the target exoskeleton parts to the preferred length along a specified axis by applying scaling factors. Upon launching the program, a graphical user interface (GUI) allows users to input scaling factors for the skeleton bones. When the "Create Model" button is activated, the program adjusts the size of the bones, along with their physical properties such as mass, mass center, and inertias. This scaling also applies to equivalent exoskeleton parts, muscle attachment points, joint locations, contact surface placements, and the initial height position. To execute the program, open a terminal, activate the virtual environment opensim-rl, install the sympy library using the command conda install sympy, navigate to the directory containing the git repository, and run the command python model_generator.py. A window will open with fields corresponding to different bones where users can input scale factors (x_factor, y_factor, z_factor) for the desired bones. It's noteworthy that there is a function for scaling PointOnLineConstraints, though its utility depends on the specific exoskeleton configuration. Notably, the y coordinate (y_factor) corresponds to the length of femurs and tibias, while the x coordinate (x_factor) corresponds to the length of feet. When scaling feet, only the x coordinate is considered for both the skeleton and exoskeleton. Scaling the head currently has no impact on the newly created model, and fine-tuning the new initial position of the pelvis's height is required, with variations expected in its positioning relative to the ground. The resulting coupled system for exoskeleton and human musculoskeletal model can be seen in Figure 6.

The primary objective was to mechanically integrate the Autonomyo exoskeleton with the musculoskeletal system following a change in the robotic devices. The process involved adjusting the inertias of various exoskeleton parts, implementing modifications and scaling to confine exoskeleton movement to the sagittal plane and ensure compatibility with the musculoskeletal model. The coupling between the two systems was then refined to emulate the real-life straps of the exoskeleton. In adjusting the inertia of exoskeleton parts, each component was initially represented by a simplified 3D geometrical object to verify the physical correctness of their inertias and prevent issues during walking simulations. Using a Python program, the dimensions of these simplified objects were determined by solving the system of equations derived from inertia tensors, subsequently compared with the true dimensions of exoskeleton parts. We then rely on an assembly error metric between the musculoskeletal model and exoskeleton. The assembly error reflects how well the model adheres to constraints governing the linkage between the two systems. In OpenSim, an assembly error below or equal to 1e-10 is considered indicative of the model complying with imposed physical constraints. With extensive automated adjustments in coupling, the resulting assembly error measured by the physics engine of the OpenSim simulator remained below 1e-10.

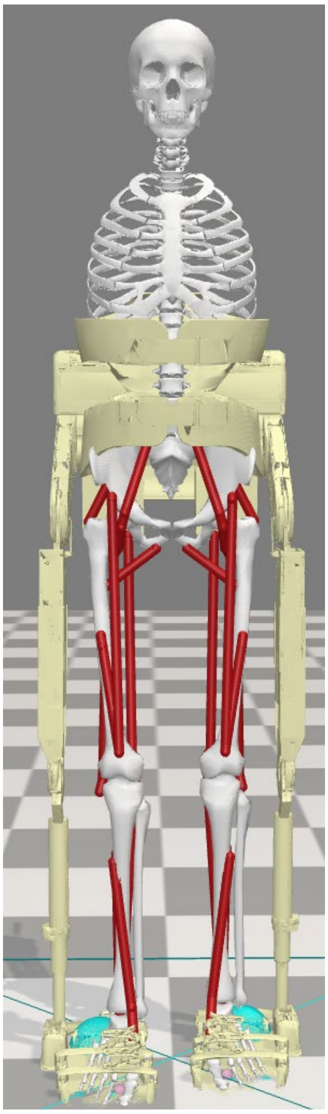

Figure 6: Autonomyo Exoskeleton Coupled with the CESPAR Musculoskeletal Model

3. Usage in the HBP Neurorobotics Platform

In this deliverable, we also provided an example of integration of the coupled human musculoskeletal system and assistive robotics device gait optimisation experiment into the HBP Neurorobotics Platform. This integration focuses on achieving seamless coordination between the reflex controller and the musculoskeletal model, enabling the exoskeleton to facilitate healthy walking patterns within the HBP Neurorobotics Platform.

The experiment employs a reflex controller, which modulates activation of the model's muscles. By leveraging this controller, a user can effectively synchronise the movements of the exoskeleton with the underlying musculoskeletal system, ensuring harmonious, natural walking patterns. This integrated approach aims to optimise gait patterns during exoskeleton-assisted locomotion. By fine-tuning the reflex controller's parameters and dynamically adjusting the model's muscle activations, a user can achieve adequate synergy between the exoskeleton and the human body. Based on our progress thus far, a user has the ability to refine the gait optimization process with comprehensive simulations and analysis. Future work could study the intricate interplay between the reflex controller and the musculoskeletal model, which could help us to unlock new insights into the biomechanics of exoskeleton-assisted walking, ultimately leading to enhanced performance, stability and user comfort.

Furthermore, by integrating the coupled gait optimisation experiment into the Platform, we provide researchers and practitioners with a powerful tool to explore and advance the field of exoskeleton-assisted locomotion. This integration should foster collaboration, enabling the exchange of knowledge and the development of novel strategies for optimising gait patterns and maximising the benefits of exoskeleton technology. We anticipate that this integration will pave the way for transformative advancements in the field, propelling us closer to the realisation of exoskeleton systems that integrate seamlessly with the human body, restoring mobility and enhancing the overall quality of life for individuals with impaired mobility.

3.1 How to conduct an experiment using the new CESPAR capability on the NRP

To run the experiment on the HBP-NRP, begin by navigating to the 'nrp' directory of the publicly available CESPAR source code (https://github.com/alpineintuition/cespar.git). Details of this integration are provided as a part of the source code documentation, as well as details of the software integration in https://github.com/alpineintuition/cespar/blob/main/md_files/nrp.md. Then, run the following commands:

```
sudo chmod +x nrp_initializer.sh
sudo ./nrp_initializer.sh
pip install git+https://github.com/stanfordnmbl/osim-rl.git
pip install deap scikit-learn mpi4py
./experiment_initializer.sh
```

You can adjust the desired simulation duration in the 'simulation_config.json' file located in the 'test_cmaes' directory in the 'nrp' directory. It is set to 10 seconds by default.

4. Optimisation and Learning Pipeline

Given the prevalence of optimisation processes throughout evolution, it is logical to approach the study of locomotion as a mathematical optimisation problem in order to emulate biological optimisation. Mathematical optimisation involves selecting the best possible solution from a feasible solution set based on a specific criterion. Typically, this involves minimising or maximising a fitness function. Several algorithms exist to handle different optimisation problems, including particle swarm optimisation (PSO), genetic algorithms and others.

Initially, a metha-heuristic optimization approach known as Particle Swarm Optimization (PSO) with a Central Pattern Generation (CPG) was employed at the project's onset. In this configuration, the controller managed the walking model, while the CPG controller learned various reflex signals. Subsequently, after a specified timeframe (typically three seconds), the CPG assumed control by generating replicas of the reflex signals. In this phase, the project aimed to optimise the CPG parameters, including frequency, amplitude, trunk angle, and phase-shift, with a focus on speed modulation and stability through PSO. However, the reported stability of the solution was deemed suboptimal for most parameters, underscoring the necessity for a more sophisticated methodology to enhance the model's stability.

4.1 CMA-ES Implementation details

In our project, we introduced a coupled neural network controller alongside the reflex controller. By using a neural network, the coupled controller model could learn and adapt to spinal reflex signals, leading to improved locomotion modulation. Since the chosen controller model includes numerous adjustable parameters, finding the optimal parameter set to maximise a given fitness function was challenging.

To tackle this issue, we employed an optimisation algorithm called Covariance Matrix Adaptation Evolution Strategy (CMA-ES) by Hansen and Ostermeier in 2001[11]. CMA-ES, as a population-based algorithm, offers a robust approach for identifying the optimal parameter set within a complex parameter space. By leveraging evolutionary principles, CMA-ES evolves a population of candidate solutions iteratively, using a covariance matrix to adapt the distribution of the solutions and guide the search towards promising regions of the parameter space. This allowed us to explore efficiently the vast solution space and discover the parameter configuration that optimises the desired fitness function.

The integration of CMA-ES within our project enabled us to fine-tune the parameters of the coupled controller and reflex controller, optimising their performance and enhancing the overall locomotion capabilities of the system. By iteratively refining the parameter values, we strove to achieve locomotion patterns that closely resemble biological optimisation

To summarise, the optimisation process employed in this project followed the Covariance Matrix Adaptation Evolution Strategy (CMA-ES). The initial population of individuals was generated by sampling from a Gaussian distribution. The width of the distribution corresponded to the estimated covariance matrix, while the centre (μ) represented an estimation of the mean of the individuals. Subsequently, at each generation, a new population was created using an updated estimate of the covariance matrix based on the present generation.

In order to select the individuals for the next generation, the best-performing individuals from the previous generation were chosen as parents, based on their fitness values. This selection process ensured that individuals with superior fitness contribute to the subsequent generations. As a result, over the course of the generation sequence, individuals with increasingly improved fitness values were generated.

To implement the CMA-ES strategy in this project, we utilised the "Distributed Evolutionary Algorithms in Python" (DEAP) framework, which serves as an evolutionary computation framework. DEAP is built upon the principles outlined by Hansen and Ostermeier in 2001, providing a robust foundation for implementing and executing the CMA-ES optimisation approach.

By employing the CMA-ES algorithm within the DEAP framework, our project achieved an efficient and effective optimisation process. This strategy allowed us to iteratively refine the parameters of the locomotion control system, driving the evolution of individuals with progressively enhanced fitness. The combination of the CMA-ES approach and the DEAP framework enabled us to explore the parameter space effectively, leading to the discovery of optimal parameter configurations that maximise the desired fitness function.

Muscle parameters and joint ranges are sourced from Geyer and Herr in 2010[12], while the parameters of the lower limbs rely on anthropometric data extracted from David A. Winter in 2009[13]. The optimization process focuses exclusively on the parameters linked to the sensors-muscles mapping, with the corresponding rules.

The optimisation process used in this deliverable involved tuning several tuneable hyperparameters to ensure the effectiveness of the optimisation algorithm. These hyperparameters included the number of generations, the number of individuals per generation, the initial width of the parameter distribution, the optimisation mode (2D or 3D), the maximum simulation duration, and the initial and target speeds of the simulation, among others. These hyperparameters played a crucial role in shaping the optimisation process and determining the quality of the resulting controller parameters.

To optimise the controller parameters, various text files were utilised, each containing the specific parameters that are subject to optimisation. In the case of the adapted reflex model, there were 37 optimisation parameters that required fine-tuning for the model to achieve robust and human-like

[11] Hansen, N. & Ostermeier, A. (2001). Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, *9*(2), 159-195.

[12] Hartmut Geyer and Hugh Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 18(3):263–273, June 2010.

[13] David A. Winter. Biomechanics and Motor Control of Human Movement. John Wiley & Sons, Inc., Hoboken, NJ, USA, September 2009.

walking behaviour. However, before applying the CMA-ES algorithm, it was necessary to scale these parameters appropriately.

The adapted model encompasses 37 optimization parameters essential for achieving robust and human-like walking in a healthy manner. For the successful implementation of CMA-ES, which initialises a multi-dimensional Gaussian with a diagonal covariance matrix, it becomes necessary to scale these parameters adequately. This scaling is facilitated by normalising the state space of the parameters to ensure uniform width for each dimension. In the code, this normalisation is achieved through the definition of a parameter range, denoted as par_space, and a linear search of finding the best hyperparameter is implemented. This range consists of two arrays representing the lower and upper bounds of the parameters, both of the same length. The circuitry involves 37 control parameters per leg, signifying the influence of reflex modules on each muscle's activation throughout the gait cycle, the trunk lean angle, and additional parameters pertaining to reactive foot placement.

To achieve parameter scaling, the CMA-ES algorithm initialised a multi-dimensional Gaussian distribution with a diagonal covariance matrix. Therefore, the state space of the parameters needed to be normalised, ensuring that each dimension had the same range. This normalisation process was implemented in the code by defining a parameter range, denoted as 'par_space' which consists of two arrays representing the lower and upper bounds of the parameters. Both arrays had the same length, and their values determined the acceptable range of each parameter.

In the model, the walking cycle is partitioned into three distinct phases (swing, stance, and stance end). The activity of each muscle during these cycle phases is determined by a linear combination of signals sourced from various inputs, including muscle length and force sensors, joint sensors, and ground sensors. This model, denoted as feedback-based locomotion, incorporates sensory modalities from muscles (such as Golgi tendon and muscle spindle), feet (pressure sensors), and the vestibular system. These modalities are represented as affine transformations of muscle length, proportionality to muscle forces, and a PD control for the vestibular system that aims to align joint angles (e.g., the trunk) with a reference angle. Muscle activity is then derived as a nonlinear combination of these diverse sensory inputs. Through the integration of these sensory modalities, a closed action-perception loop is established, embodying a closed-loop control architecture that emulates the motor control system found in the human body.

The controller parameters under consideration included 37 control parameters per leg. These parameters govern the contributions of the reflex modules to each muscle's activation throughout the gait cycle. Additionally, the controller parameters include the trunk lean angle and parameters related to reactive foot placement. By appropriately adjusting these parameters through the optimisation process, the model can achieve a more refined and natural walking pattern that emulates human locomotion.

A fitness function serves as an objective measure determining the proximity of a given design solution to achieving predefined goals. In this context, the candidate solution is the input to the problem, and the resulting output signifies the fitness of this candidate solution concerning the optimization objectives. The simulation continues until the human model completes 10 seconds (i=1000) or when the pelvis descends below 0.6 metres. Throughout these simulations, various costs are accumulated, including a survival cost for each timestep i (with an initial weight of 0.1), a footstep cost whenever a new footstep is taken (with an initial weight of ten), and a success cost (with an initial weight of one). The total cost is thus characterised by these components.

$$
\begin{aligned}
J(\pi) &= R_{alive} + R_{step} + R_{success} \\
&= \sum_i r_{alive} + \sum_{step_i} \left(w_{step} \cdot r_{step} - w_{vel} \cdot c_{vel} - w_{eff} \cdot c_{eff} \right) + R_{success}
\end{aligned}
$$

In the context of this equation, the comprehensive fitness function is the sum of three distinct cost functions. Firstly, R_alive assesses whether the model remains 'alive' throughout the entire simulation, denoted by the model staying above 0.6 metres. Secondly, R_step, the footstep cost, evaluates the model's step behaviours rather than its instantaneous actions. Lastly, R_success examines whether the model avoids falling for the entire simulation duration, typically 10 seconds.

R_step is designed to be high when the model walks with minimal effort at target velocities, incorporating a step cost to prevent the model from getting stuck (w_step · r_step), a penalty component (w_vel · c_vel) for deviations from the target velocity, and an effort penalty (w_eff · c_eff) to discourage excessive effort.

The outcome from the optimization trial reveals a convergence of the error, falling below the specified acceptance criteria of 0.05. To enhance visualisation, the error in the y-axis is presented in log-scale. The x-axis corresponds to the number of evaluations, encompassing the entirety of the optimization process, where a convergence of the error is achieved approximately after 175 evaluations, see Figure 7.

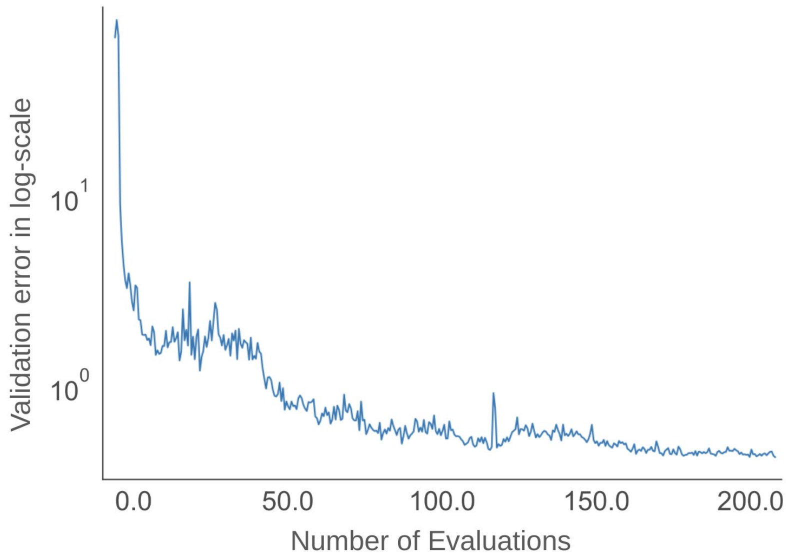

Figure 7: Convergence of error in log-scale.

The coupled reflex model exhibits joint angles that closely resemble those observed during human walking, as demonstrated in Figure 8. To achieve this similarity, the joint angles throughout the entire gait cycle were normalised to a standardised distribution consisting of 100 data points, representing 0% to 100% of the stride. This normalisation process allows for a direct comparison between the model's joint angles and the corresponding joint angles observed in humans walking at speeds ranging from 1.00 m/s to 1.25 m/s, commonly referred to as free speed. Figure 8 shows the hip joint angles obtained from 7 experiments with our modified reflex controller compared against healthy human hip joint trajectory during walking. Initially, the analysis focuses on the first three experiments, where the sole variation lies in the initial width of the probability distribution, termed "init," aiming to scrutinise its impact. A wider init is noted to provide increased flexibility in parameter selection due to the expanded width of the probability distribution. Subsequently, Experiment 4 and subsequent iterations involve diverse initial and target speeds, exploring the model's capability to attain the specified speed during optimizations. In the subsequent experiments, both initial and target speeds are progressively reduced, culminating in Experiment 7's objective of achieving a velocity of 0.5 m/s. Notably, the initial speed differs between Experiments 4 and 7, as maintaining a significant disparity between initial and target velocities resulted in model capabilities to avoid a fall. All results were normalised over one gait cycle. The vertical red line indicates the end of the stance phase.

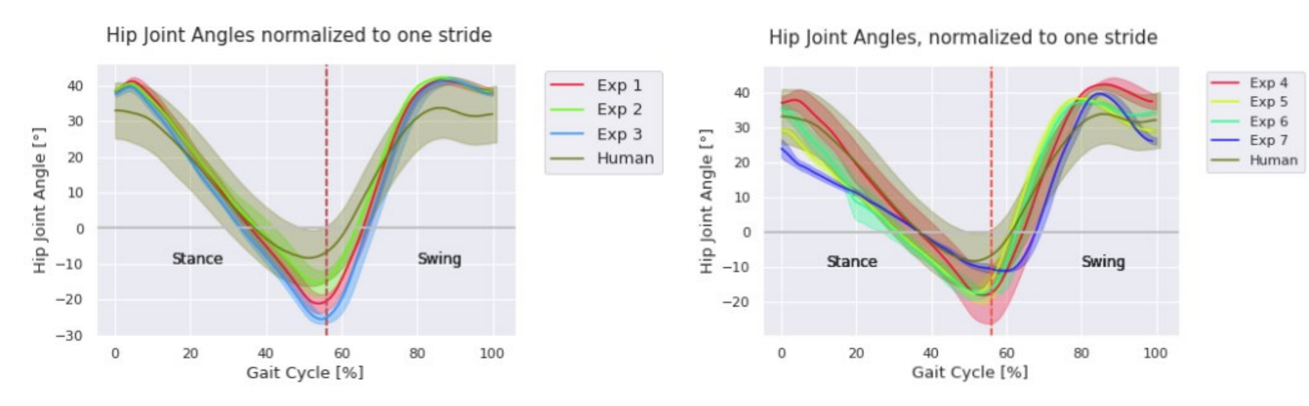

Figure 8: Hip Joint Angles for seven experiments with the RFX-model.

In the context of various experiments employing the model, the focus centres on the hip joint angles, which have been normalised across a single gait cycle. The presentation distinguishes between two sets of experiments: Experiments 1 to 3 are depicted in the left figure, while Experiments 4 to 7 are represented in the right figure. Notably, a vertical line is incorporated in the visualisations to signify the conclusion of the stance phase. In order to analyse the effectiveness of the combined coupled reflex controller, a series of optimisation experiments were conducted within the OpenSim environment. Specifically, we focused on seven experiments, each with a simulation duration of 10 seconds. To ensure a smooth gait initialisation, the coupled reflex controller was activated after 0.4 seconds into the simulation. Convergence of the optimisation process was ensured by using a large number of generations, and multiple repetitions of each experiment were performed to validate the reproducibility of the results. The parameter space consists of the 37 parameters from the reflex model and 44 parameters from the coupled controller (four centres, four variances and 36 weights). In the seven experiments conducted, we also varied the weightings or relative importance of the parameters used in the coupled controller model. It was observed that a proximo-distal gradient in neuromechanical control leads to more natural locomotion. Through these optimisation experiments, we aimed to assess the performance and similarity of the combined coupled reflex controller, leveraging insights from the literature to guide our analysis. The diverse range of experiments allowed for a comprehensive evaluation of the controller's behaviour and its ability to generate human-like locomotion patterns. The connection weights after the optimisation and learning framework are given in Figure 9. The figure shows weight matrices used in two different experiments. In the first (left), the target speed was set to 1.3 m/s while in the second (right), it is set to 1.5 m/s. Each column of the weight matrices represents a motor primitive and each row represents a muscle activation. The connection weights between muscle activations and motor primitives are set to be only positive. The darker the square colour, the more positive the weight.

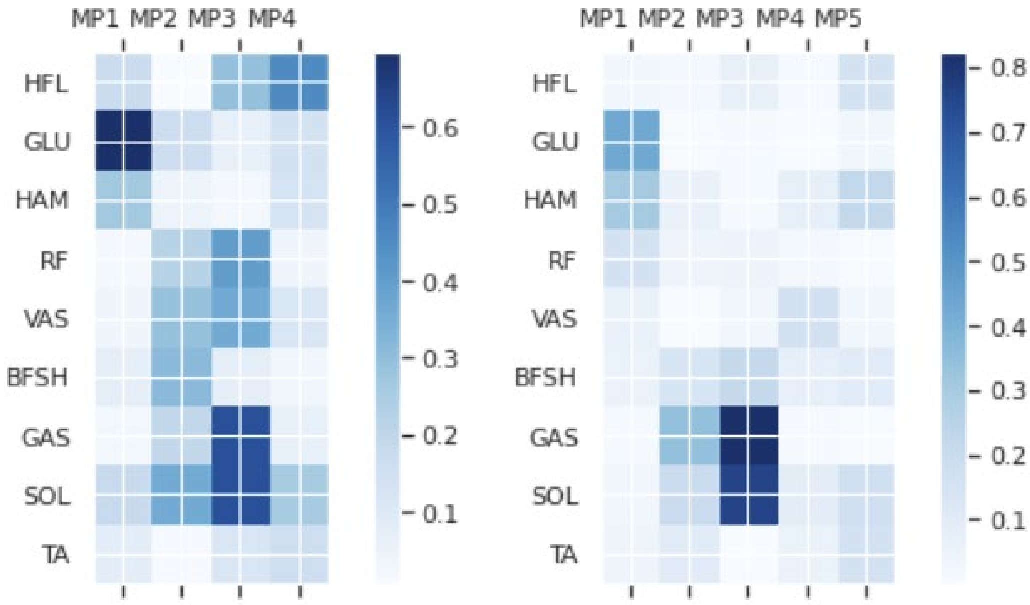

Figure 9: Weight matrix used for experiments with a target speed of 1.3 m/s, and 1.5 m/s

4.2 Deep neural network and implementation details

We also integrated a neural network implementation with reinforcement learning to obtain the actuations in the exoskeleton joints. The state space for the Reinforcement Learning (RL) implementation is delineated by the length and velocity of the contractile elements in the muscle model mapped to the joint angles of the exoskeleton. The output of the neural network serves as the stimulus vector directed to the activation dynamics of the exoskeleton actuators, governed by a first-order differential equation that incorporates neural delay. The transition probability is articulated through the probabilistic policy function alongside a continuous reward function. Various approaches exist for formulating reward functions in RL problems, ranging from high-level goals like achieving forward motion to reaching specific positions in joint space. Alternatively, reward functions can be engineered as combinations of objectives, such as minimising energy while attaining a goal position. In this approach, akin to imitation learning problems, the reward is defined as a metric indicating the proximity of the musculoskeletal system's state to a given joint trajectory obtained through human gait data. Consequently, policy search is not framed as a high-level goal, like a position-specified reward function, but rather as the imitation of a desired trajectory used as a reward function in the RL formulation.

As a result, the reward function is articulated as a sequence of joint positions and velocities for the coupled system, structured to minimise the disparity between the provided and actual trajectories. Formulating the reward function in this manner, emphasising the minimization of motion trajectories and their disparities, aligns with the principles of an inverse Reinforcement Learning (RL) formulation. However, it is acknowledged that determining a comprehensive reward function capable of solving the entire inverse RL problem exceeds the scope of this deliverable. Notably, human movements are characterised by the creation of smooth trajectories with minimal tremors. In order to encourage the Deep RL solution to exhibit such smoothness, an additional term penalising high acceleration is integrated into the reward function as a regularisation component, aimed at reducing trembling movements and promoting overall smoothness. In summary, the reward function is expressed as the sum of weighted differences between desired and actual trajectories, encompassing not only position differences but also velocity differences.

$$r_t = \sum_{i=0}^{N} w_{q,i}\left(-\left\|q_{d,i} - q_{o,i}\right\|^2\right) \quad + \sum_{i=0}^{N} w_{\dot{q},i}\left(-\left\|\dot{q}_{d,i} - \dot{q}_{o,i}\right\|^2\right) \quad + \sum_{i=0}^{N} w_{\ddot{q},i}(\ddot{q}_{o,i})^2$$

The deep neural network employed in this study adopts an actor-critic architecture and makes necessary extensions to suit coupled control problems. Both the actor and critic networks utilise feedforward layers, comprising five hidden layers for the actor and three for the critic. The neurons in the critic network's layers are activated using PReLu (Parametric Rectified Linear Unit), with a linear layer conveying the network's output to a single value representing the value function. A similar design principle is applied to the actor network, where the neurons in the hidden layers are activated by the Tanh function. A crucial aspect of the actor network lies in its last layer, responsible for providing the probabilistic distribution of the action values.

This deliverable delves into the examination of a diverse range of movement generation within a musculoskeletal system and exoskeleton control, leveraging the CMA-ES and deep reinforcement learning. Multiple experiments of varying complexity are conducted to evaluate the efficacy and quality of the introduced learning and optimization framework. In comparison and also to validate our approach to assess the quality, versatility, and readiness of our solution, we conducted experiments that are based on data derived from human walking contrasted with information extracted from the solution of our experiments. Figure 10 depicts the comparison of joint angles between human data (grey area) and the model, revealing higher correlation values in hip, knee and ankle flexions. The shaded region depicts the range within which various joint angles are expected to fall, representing the joint angle values of a healthy individual. The percentage indicated alongside each joint angle denotes the extent to which the model's values align with the healthy range. A total of 9 steps were executed, each having an average length of 1.59m and a duration of 1.47s. The alignment between the human data and the joint trajectories of the exoskeleton indicates the resulting controller allows users to reach a healthy gait pattern, see Figure 11. Our optimization and learning framework provides users with the flexibility to interact with the solution according to their specific needs. The solution itself encompasses a comprehensive toolchain, allowing users to integrate an exoskeleton, connect it with a human musculoskeletal model, personalise its scaling based on individual data, and subsequently engage in the optimization of the control architecture tailored to specific subject requirements. While acknowledging the current solution's need for thorough clinical validation, preliminary outcomes indicate that a requisite level of validation has been achieved. This prompts the initiation of a clinical study to further establish the viability of the solution for real-world application in patient settings. It's worth noting that the validation process in the highly regulated MedTech field requires additional scrutiny, extending beyond the scope of the existing deliverable.

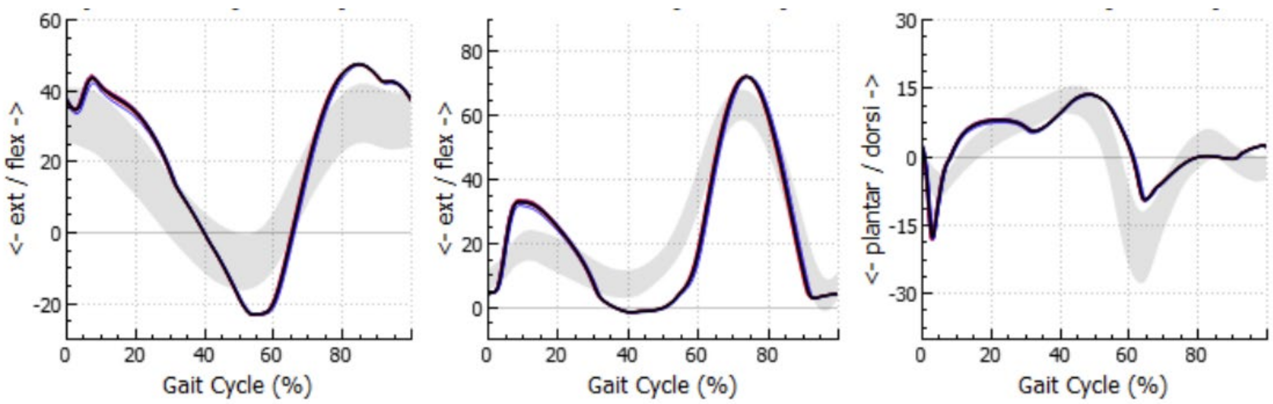

Figure 10: Comparison between human data and exoskeleton, gait cycle (Hip, Knee, Ankle)

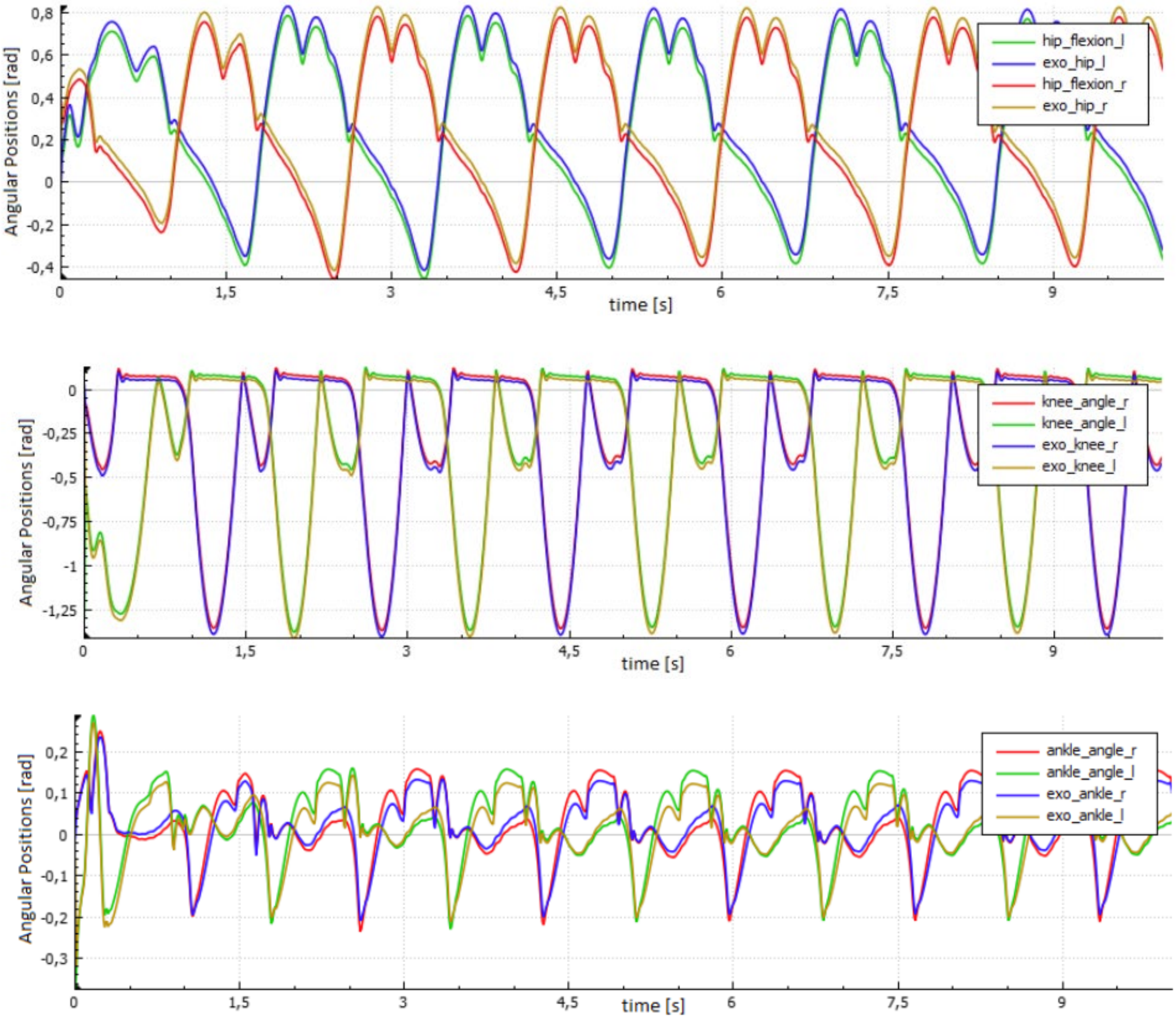

Figure 11: Comparison between human data and exoskeleton, joint trajectories (Hip, Knee, Ankle)

4.3 Summary of work performed

Based on our *in-silico* optimisation and learning pipeline, we presented a collaborative effort between Alpine Intuition and Autonomyo to build a complete workflow of an-silico experimentation environment of assistive robotics and human musculoskeletal systems. We demonstrated the implementation of the models as controllers in various scenarios. The exoskeletons are intended to assist abnormal gaits of subjects with a continuous interaction in a comfortable way. To address this, we proposed and implemented *in-silico* experiments with neural network-based controllers by employing computational models and leveraging state-of-the-art optimisation and learning techniques. The implementation of our *in-silico* experiments effectively validated our workflow replicating both healthy individuals as well as subjects with spinal cord injuries (SCI) presenting various abnormalities as an ongoing effort. These initial but promising outcomes have implications for leveraging the innate dynamics of musculoskeletal dynamics of the human motor control system to expand and adapt the design principles of exoskeletons as wearable devices. The outcome of our deliverable also allows our users to validate the actuation of exoskeletons capable of adapting to changing environmental conditions.

5. Looking Forward

In this deliverable, we aimed to help overcome the challenges of exoskeleton adaptation for SCI patients by building an extensive in-silico experimentation environment for predictive neuromechanical simulations and neurorehabilitation. Our goal was to provide predictive simulations for exoskeleton and prosthetic usage in neurorehabilitation. The idea was for the simulation to produce motions and desired behaviours without directly involving human subjects and rehabilitation assistants. These studies aim at better understanding the effect of prosthesis or orthosis in human subjects. Successfully predicting motions that reliably fit human subjects' needs while using exoskeletons/exosuits could be used to accelerate the rehabilitation treatments of the patients while reducing the burden of experiments on the patients and rehabilitation assistants. Our predictive simulation environment will also allow researchers to study human motor control models in a realistic virtual environment.

The CESPAR exoskeleton simulator will allow users to exploit the capabilities of HBP-NRP with the integration of exoskeletons/exosuits/prosthetics into the human musculoskeletal simulations. The major advantage of HBP-NRP for exoskeleton simulations is its core design principle of closed-loop action-perception simulation. With the incorporation of our CESPAR simulator, the HBP-NRP will allow users to validate brain and spinal cord models and directly test them in clinical applications, either with or without assistive devices and in different rehabilitation schemes. Our simulator will allow interested communities to extend this research, while also making an exoskeleton simulation readily available for HBP partners.

Furthermore, we hope that our work will highlight that modelling, predicting and assisting human locomotion is a complex and useful undertaking, worthy of public attention and funding as it should help to accelerate the development of robotic solutions for people suffering from motor control diseases and people affected by injuries that require short- and long-term rehabilitation.