Implementation of 3D Path Control for Stair Scenarios

Marine Moutarlier
Microtechnique-Robotics
EPFL, Lausanne, Switzerland
marine.moutarlier@epfl.ch

Zeynep Özge Orhan
REHAssist
EPFL, Lausanne, Switzerland
zeynep.orhan@epfl.ch

Mohamed Bouri
REHAssist
EPFL, Lausanne, Switzerland
mohamed.bouri@epfl.ch

***Abstract*—This work presents the implementation of a 3D path control for stair ascent and descent scenarios using Autonomyo exoskeleton, designed to assist individuals with lower limb disabilities. Effective path control mechanisms are crucial for enhancing mobility and independence in users, particularly during complex motor tasks such as stair climbing. This report details the hardware setup, data preprocessing steps, and the application of mapping and finite state machine (FSM) approaches to manage transitions between different gait phases, ensuring accurate and reliable assistance. The results demonstrate the effectiveness of the proposed method. Experimental evaluations validate the system's performance, showing its potential to provide meaningful assistance in real-world scenarios.**

I. Introduction

Stair climbing is a complex motor task that requires precise coordination and strength. Implementing effective stair-climbing/descending mechanisms in exoskeletons is crucial for enhancing mobility and independence in users, particularly those with lower limb impairments. Implementing stair ascent and descent ensures that the user feels supported, leading to a more natural and effective gait. Effective path control mechanisms are essential for safe and efficient stair ascent and descent. This involves precise control algorithms that can adapt to user dynamics, ensuring stability and support throughout the motion.

A. Related Work

Path control in rehabilitation robotics is a critical concept for facilitating patient-cooperative training. In the study by Duschau-Wicke et al. [1], a method is proposed that allows the robotic system to adjust the assistance level based on the patient's own movements. This approach ensures that the rehabilitation process is adaptive and responsive to the patient's needs. Similarly, the work by Martínez et al [2] introduces a velocity-field-based controller designed for a lower limb exoskeleton, focusing on assisting leg movements during walking. This controller uses a virtual flow field to guide the movement of the hip and knee joints, enabling users to maintain a natural gait while receiving necessary support. The study by Martínez et al [3] introduces a lower limb exoskeleton control approach that coordinates hip and knee movements during the swing phase of gait. It allows users to control step time through time-invariant constraints and step length via real-time path planning. Together, these studies highlight the importance of adaptive control strategies in robotic-assisted gait training. This work is about extending those methods to stair gait.

II. Methods

A. Hardware and path control

The Autonomyo exoskeleton (Appendix,Fig. 40) is designed to enhance lower limb mobility by providing targeted assistance through a path control algorithm. It has three active degrees of freedom (hip flexion/extension, knee flexion/extension, and hip abduction/adduction) actuated by brushless motors with integrated torque sensors, and three passive degrees of freedom at the ankle.

The system employs an impedance controller with an inner torque control loop to dynamically adjust assistance based on the user's movements and load conditions, which are measured by sole-embedded sensors. The controller used is a path controller that computes a linear interpolation of three points to fit a curve. The measurements of the exoskeleton are matched to the closest point on this curve, referred to as the target point. The segments of the exoskeleton are adjustable. The implementation of the path controller is detailed in Appendix, Fig. 41.

B. Utilized datasets

The first step of this implementation involved identifying datasets in the literature that focused on stair ascent and descent gaits. Data that included three specific joint angles: hip flexion/extension, knee flexion/extension, and hip abduction/adduction was sought. In total, three open-datasets were found. One was from Losing et al. (2022) [4], another from Gregg et al. (2021) [5], and the third was obtained from our own measurements using Xsens on the exoskeleton.

The primary dataset from Xsens includes data for one subject. The Losing dataset provided data from 10 participants, and gait at three distinct speeds: fast, slow, and self-selected. Similarly, the Gregg dataset included data from 5 participants. All datasets were three-dimensional, following conventions were chosen: the x-axis represents hip flexion/extension, the y-axis represents knee flexion/extension, and the z-axis represents hip adduction/abduction and all included stair ascent and descent. This consistent framework was maintained throughout the entire implementation.

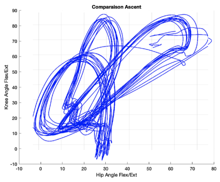

Fig. 1: Xsens data with walking, SA and SD cycles

The same procedure was applied to all datasets. The data was segmented into cycles, isolating the Stair Ascent and Stair Descent from the rest of the dataset, as illustrated in Fig. 1. This allowed to obtain closed curves representing the different gaits. Data from multiple individuals was collected, then averaged it into a single cycle, and finally the resulting curves were smoothed.

C. Curve fitting

The data processing involved a two-step procedure to refine the gait cycle. Initially, a polynomial fit was applied to the raw data points, see Fig. 2. A second-degree polynomial was fitted using MATLAB's polyfit function, as defined in Eq. 1.

$$\mathrm{p} = \mathrm{polyfit}(x, y, 2) \text{ with } p(x) = p_1x^2 + p_2x + p_3 \qquad (1)$$

This command returns the coefficients of a polynomial $p(x)$ of degree 2 that best fits the data y in a least-squares sense. The fitted polynomial provided a smooth representation of the data, capturing the underlying trend effectively. This was done to allow the controller to be more efficient and avoid a jumpy behavior. Following the polynomial fitting, a Gaussian smoothing filter was applied to reduce noise. The resulting data is illustrated in Fig. 3.

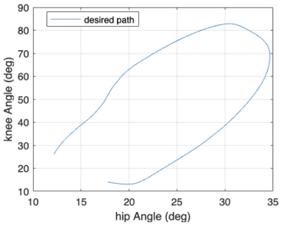

Fig. 2: 2D Literature data (Xsens) for SD before curve fitting

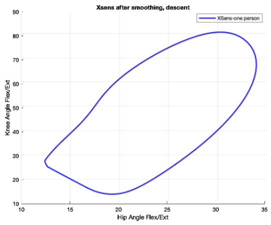

Fig. 3: 2D Literature data (Xsens) for SD after curve fitting

D. Data after preprocessing steps

The same preprocessing steps were applied to the other datasets, with different approaches being tested as needed. After completing these preprocessing steps, a total of 15 datasets were evaluated on the exoskeleton. These datasets are illustrated in Fig. 4 to Fig. 7.

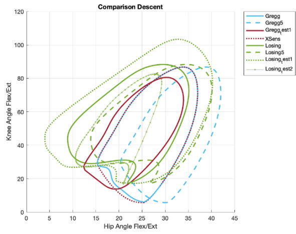

Fig. 4: Literaure data for SD in 2D

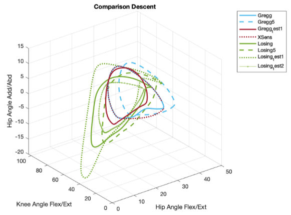

Fig. 5: Literaure data for SD in 3D

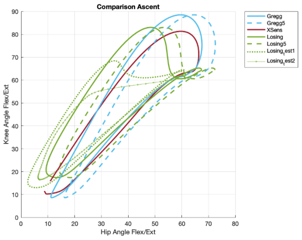

Fig. 6: Literaure data for SA in 2D

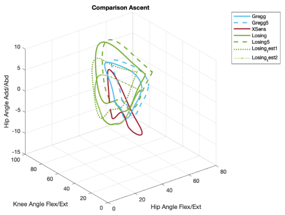

Fig. 7: Literaure data for SA in 3D

E. Results after preprocessing steps

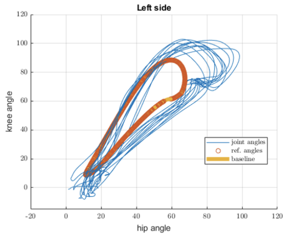

Fig. 8: Gregg's data for SA for a 2D visualisation

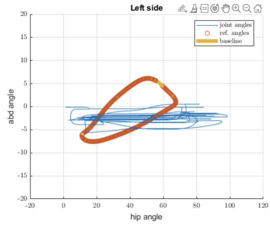

Fig. 9: Gregg's data for SA for a 3D visualisation

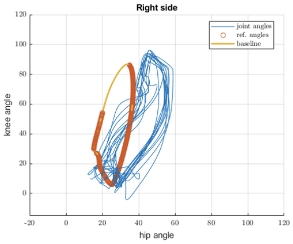

Fig. 10: Gregg's data for SD for a 2D visualisation

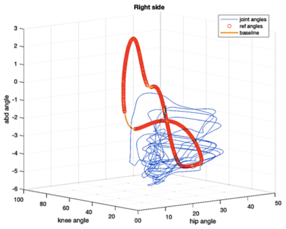

Fig. 11: Gregg's data for for a 3D visualisation

Using the encoders instead of the IMU has resulted in unsatisfactory hip abduction angles. An investigation was conducted to better understand the underlying issues. Gregg's results were found to be the most reliable, while the other datasets were discarded as they were significantly different from Autonomyo's measurements.

F. Investigating the results

There was a difference between the measured angles and the baseline from the literature. This discrepancy shows because the exoskeleton may have slightly different definitions or design variations, leading to an error between the measurements and the baseline. To investigate the source of this error, the relative error between the measurements and the baseline was plotted, as shown in Fig 12.

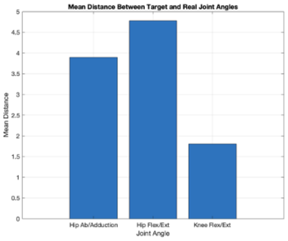

Fig. 12: Averaged relative errors between measurements and baseline for SD, for each angle

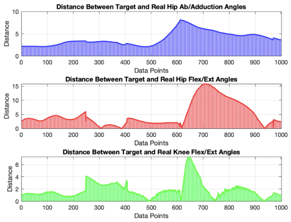

Fig. 13: Relative errors between measurements and baseline for SD, throughout a cycle

G. IMU filtering for hip AA

The measurment of hip abd to IMU was changed to reduce the large error seen in Fig. 12.IMU is known to be a noisy measurement, a filter is applied, as seen in Fig. 14.

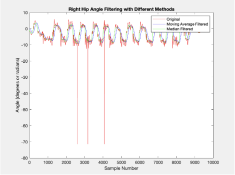

Fig. 14: IMU data filtered

The moving average filter is applied to smooth the data. The filter is defined by the following equations:

$$y[n] = \frac{1}{N} \sum_{k=0}^{N-1} x[n-k] \tag{2}$$

where N is the window size, $x[n]$ is the input signal, and $y[n]$ is the filtered signal.

In this case, the window size N is set to 150. This was done on both left and right hip abduction/adduction angle (`angleHipAA`), the filtered signal is given by:

$$\text{AngleFiltered}[n] = \frac{1}{150} \sum_{k=0}^{149} \text{angleHipAA}[n-k] \tag{3}$$

This allowed to strongly reduced the noise on the IMU measurements. This was tested on the exoskeleton.

H. Second tests results

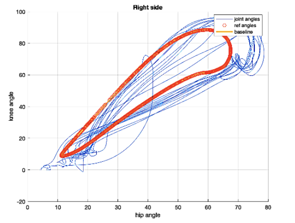

Fig. 15: Gregg's data for SA for a 2D visualisation

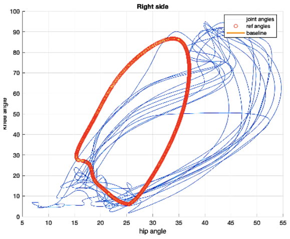

Fig. 16: Gregg's data for SD for a 2D visualisation

The use of the IMU was helpful, now more points have found a match from the baseline. But the results indicate a significant error and shift between the measured data and the baseline still. This discrepancy is particularly evident in the stair descent (SD) data (Fig. 16), highlighting the necessity for implementing a mapping.

I. Hip flexion/extension mapping

The relative error remains the most significant in hip flexion/extension for both Stair Ascent (SA) and Stair Descent (SD). Following the study by A. Dal Prete, *An Online Method for Locomotion Modes Recognition in Wearable Lower-Limb Exoskeletons* [6], a mapping of the hip flexion/extension was implemented.

Two types of mapping were implemented:

1) A mapping for SA
2) A mapping for SD

The goal of this mapping is to adjust the measurement angles of the exoskeleton to more closely match the given baseline. The function used for the respective mapping is defined in Eq. 4.

$$\phi(x, \dot{x}, \ddot{x}) = [1, x] \tag{4}$$

The weights were defined as : For Stair Ascent (w_{SA}) and Descent (w_{SD}) :

$$w_{\text{SA}} = [2.91, 0.883] \; w_{\text{SD}} = [8.99, 0.605] \tag{5}$$

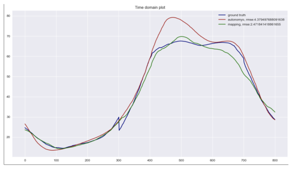

Fig. 17: Mapping for SA

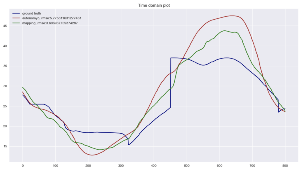

Fig. 18: Mapping for SD

The Fig. 17 and18, compares the ground truth data with two different data: "autonomyo measurements" and "mapping". The effectiveness of the mappings is assessed by examining how closely they follow the ground truth data, as indicated by their respective root mean square error (RMSE) values. RMSE is computed from the measurement to the ground truth, which is the baseline, as follow:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{6}$$

where:

- n is the number of data points,
- y_i represents the observed values (ground truth),
- $\hat{y}_i$ represents the predicted values (mapping or autonomyo)

The green mapping shows a closer adherence to the ground truth compared to the red autonomyo measurements, particularly in regions where the data exhibits significant changes. Quantitatively, for example in Fig. 18, the autonomyo measurement has an RMSE of 4.38, while the mapping has a lower RMSE of 2.47. A lower RMSE value indicates a better fit to the ground truth data, suggesting that the mapping is effective. The same observation could be seen in Fig. 17, the RMSE going from 5.77 to 3.60 for the stair descent mapping. The mapping was then implemented into the code and made accessible through a syncvar variable that could be changed during the experiments. The pseudocode is the following:

Algorithm 1 PathControl Mapping Functions

```
function MAPPINGSA(x)
    $x \leftarrow 2.91 + 0.883 \cdot x$
    return $x$
end function
function MAPPINGSD(x)
    $x \leftarrow 8.99 + 0.605 \cdot x$
    return $x$
end function
if useMappingSA then
    $leftHipPosition \leftarrow$ MAPPINGSA($leftHipPosition$)
    $rightHipPosition \leftarrow$ MAPPINGSA($rightHipPosition$)
end if
if useMappingSD then
    $leftHipPosition \leftarrow$ MAPPINGSD($leftHipPosition$)
    $rightHipPosition \leftarrow$ MAPPINGSD($rightHipPosition$)
end if
```

J. Results of the mapping implementation

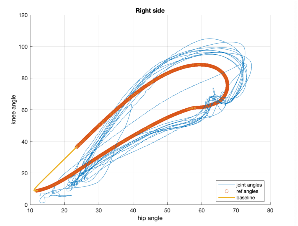

Fig. 19: Gregg's data for SA for a 2D visualisation with mapping

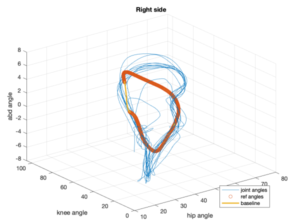

Fig. 20: Gregg's data for SA for a 3D visualisation with mapping

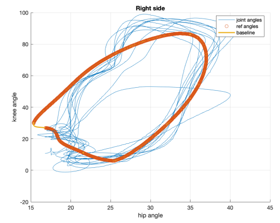

Fig. 21: Gregg's data for SD for a 2D visualisation with mapping

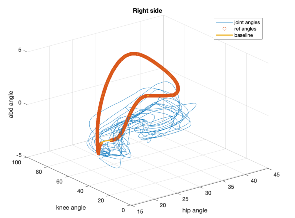

Fig. 22: Gregg's data for SD for a 3D visualisation with mapping

The mapping is particularly effective for stair descent (SD), demonstrating a significant improvement in matching the baseline data compared to the results without mapping (see Fig. 16). This indicates a successful implementation for SD. However, for stair ascent (Fig. 15) a substantial portion of the points from the baseline are not selected because they are not the closest points. This discrepancy highlights the need for implementing a finite state machine (FSM) to ensure that the correct points are consistently picked.

K. Finite State Machine implementation

As seen in the results section (II-J), the mapping was not very effective on the stair ascent data. This issue arises because the bottom corner of the data, particularly the transition from swing to stance in Fig. 24, is very sharp. The path controller operates as a closest point, which can lead to errors when the corner is sharp. In such cases, if the measurement slightly deviates from the baseline, the controller might select the incorrect point, potentially picking the wrong side of the data (example swing when in stance) and resulting in poor assistance. This problem is evident in Fig. 19.To address this issue, a finite state machine was implemented.

A finite state machine (FSM) is a computational model used to design algorithms and control systems. It consists of a finite number of states, transitions between these states based on inputs or events, and actions associated with the transitions or states.This FSM follows what was done in *A Velocity-Field-Based Controller for Assisting Leg Movement During Walking With a Bilateral Hip and Knee Lower Limb Exoskeleton* by Andrés Martínez, [2]

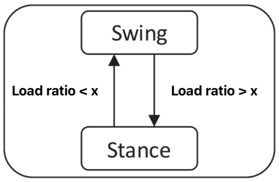

Fig. 23: FSM implementation

By clearly defining the states and transitions, the FSM ensures that the exoskeleton provides appropriate assistance at each phase of movement. This FSM has two states: swing and stance. Transitions between these states are triggered by the force detected on the load cells on the sole of the exoskeleton.

If it exceeds a value then weight is applied and the user is in stance, if not it is in stance. This structured approach ensures that the correct points are selected.

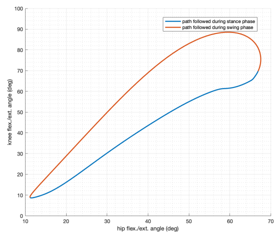

Fig. 24: FSM on SA for a 2D visualisation

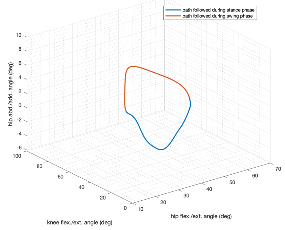

Fig. 25: FSM on SA for a 3D visualisation

Figures 24 and 25 illustrate the two different states of the finite state machine (FSM). The plots depict two phases: the swing phase (in orange) and the stance phase (in blue).

These visualizations highlight the transitions and states during the stair ascent movement.. These phases were identified by selecting points from the baseline where the force applied on the load cells exceeded a defined threshold of 40N. The baseline was separated into stance and swing phases and ordered as swing followed by stance. The code was updated to ensure that only the closest points from the stance phase are considered when in stance, and only the closest points from the swing phase are considered when in swing. This ensures that the correct points are selected, making the implementation successful and providing effective assistance.

The algorithm was implemented as follows:

Algorithm 2 FSM function

```
function FSM(x)
    if $index < 100$ then
        $n \leftarrow$ length($list\_swing$)
        $list\_swing[n] \leftarrow element$
    else
        $m \leftarrow$ length($list\_stance$)
        $list\_stance[m] \leftarrow element$
    end if
    if $left\_load\_ratio > 0.4$ then
        left_side ← stance
        right_side ← swing
    else
        left_side ← swing
        right_side ← stance
    end if
end function
```

L. Results of the FSM

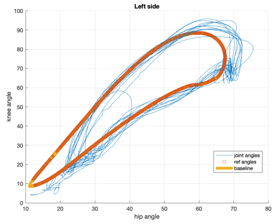

Fig. 26: Gregg's data for SA for a 2D visualisation with FSM

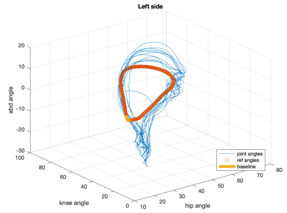

Fig. 27: Gregg's data for SA for a 3D visualisation with FSM

The implementation of the finite state machine (FSM) was successful. Compared to Fig. 19, all points from the baseline are now correctly selected.

III. Results

Since the implementation is now working, systematic experiments were conducted on three different subjects, both for stair ascent and descent. The participants characteristics were the following:

TABLE I: Participant Characteristics

Characteristic	**S1, S2, S3**
Age	30, 23, 22
Gender	Female, Female, Female
Weight [kg]	65, 75, 65
Height [cm]	171, 175, 170
Thigh Length [cm]	42, 46, 44
Shank Length [cm]	49, 52, 50
Shoe size	39, 42, 39
Dominant leg	Right, Right, Right

The results were obtained under various testing conditions and summarized in this table II. Experiments with different impedances were conducted twice to evaluate the exoskeleton's assistance effectiveness. Testing various impedances demonstrated how well the exoskeleton supported the user, with higher impedances providing better assistance by more effectively guiding the user to the desired position.

TABLE II: Testing conditions

Characteristic	**Test 1, Test 2 and 3, Test 4 and 5**
Mode	Zero-Torque , Low-impedance, High-Impedance
Impedance	0, 1 and 0.3, 2 and 0.5
Use hip IMU	yes, yes, yes
Use mapping	yes, yes, yes
Use FSM	yes for SA no for SD, ,

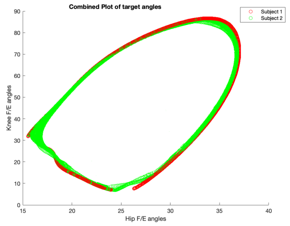

Fig. 28: Final experiments zero-torque SD for a 2D visualisation

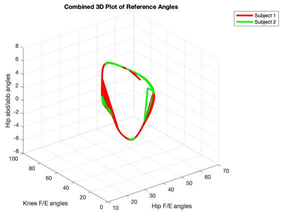

Fig. 29: Final experiments zero-torque SD for a 3D visualisation

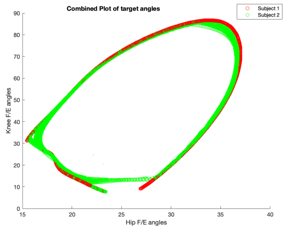

Fig. 30: Final experiments low impedance SD for a 2D visualisation

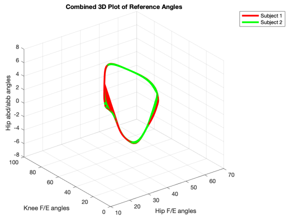

Fig. 31: Final experiments low-impedance SD for a 3D visualisation

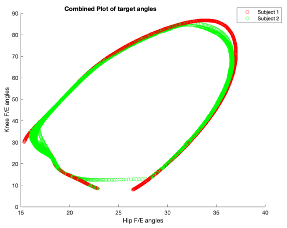

Fig. 32: Final experiments high impedance SD for a 2D visualisation

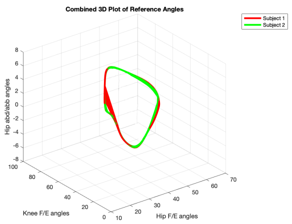

Fig. 33: Final experiments high impedance SD for a 3D visualisation

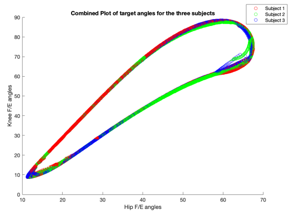

Fig. 34: Final experiments zero-torque SA for a 2D visualisation

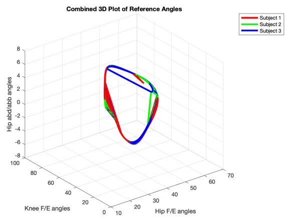

Fig. 35: Final experiments zero-torque SA for a 3D visualisation

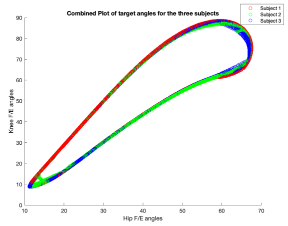

Fig. 36: Final experiments low impedance SA for a 2D visualisation

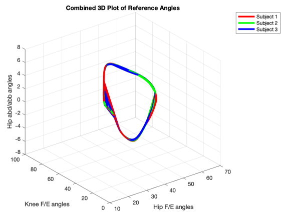

Fig. 37: Final experiments low impedance SA for a 3D visualisation

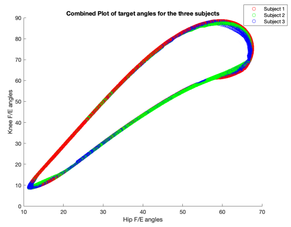

Fig. 38: Final experiments high impedance SA for a 2D visualisation

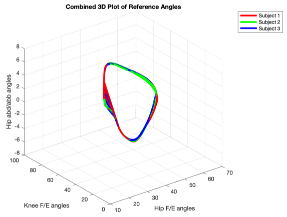

Fig. 39: Final experiments high impedance SA for a 3D visualisation

The results indicate that the path control successfully attained all target points, as evidenced by the fully closed curves for all subjects in the experiments. This confirms the effectiveness of the stair ascent and descent implementation. Specifically, for stair ascent, Fig. 39 shows that high impedance performs slightly better than the other settings. For SD it seems like Zero-Torque mode Fig. 28 has the best results. The subjects also reported a good general feeling during the experiments. More details about the scores for each categories can be found in the appendix in V-C.

IV. Discussion

The implementation of the 3D path control algorithm for stair ascent and descent scenarios using Autonomyo has shown promising results. However, several challenges were encountered throughout the study. The application of mapping techniques to align the exoskeleton's movements with the baseline data improved the performance for stair descent. However, the mapping for stair ascent did not give equally satisfactory results. This issue necessitated the implementation of a FSM to ensure a more reliable close point selection. This implementation successfully addressed this problem, enabling the controller to select the correct points, improving the overall performance for SA.Future work could focus on refining the control algorithms and expanding the scope of testing to include a broader range of experiments. The flow-controller implemented could also be tested with the stair ascend and descend.

V. Conclusion

In this work, an implementation of a 3D path control for stair ascent and descent scenarios using the Autonomyo exoskeleton, designed to assist individuals with lower limb disabilities was done. The experimental results demonstrate the effectiveness of the proposed method. The implementation of mapping functions and the finite state machine significantly reduced the root mean square error (RMSE) between the measured angles and the baseline data, indicating a better fit and more accurate assistance. Systematic experiments were conducted on multiple subjects under various conditions. The findings validate the system's performance, highlighting its potential to provide assistance in real-world scenarios. The participants reported positive outcomes underscoring the practical benefits of the implemented control mechanisms. Overall, this study contributes to the advancement of exoskeleton technology by providing a solution for stair ascent and descent.

References

[1] A. Duschau-Wicke, J. von Zitzewitz, A. Caprez, L. Lunenburger, and R. Riener, "Path control: A method for patient-cooperative robot-aided gait rehabilitation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 1, pp. 38–48, 2010.

[2] A. Martínez, B. Lawson, C. Durrough, and M. Goldfarb, "A velocity-field-based controller for assisting leg movement during walking with a bilateral hip and knee lower limb exoskeleton," *IEEE Transactions on Robotics*, vol. 35, no. 2, pp. 307–316, 2019.

[3] A. Martínez, B. Lawson, and M. Goldfarb, "A controller for guiding leg movement during overground walking with a lower limb exoskeleton," *IEEE Transactions on Robotics*, vol. 34, no. 1, pp. 183–193, 2018.

[4] V. Losing and M. Hasenjäger, "A multi-modal gait database of natural everyday-walk in an urban environment," *Sci Data*, vol. 9, p. 473, 2022.

[5] S. Cheng, E. Bolivar-Nieto, and R. D. Gregg, "Real-time activity recognition with instantaneous characteristic features of thigh kinematics," *IEEE Trans Neural Syst Rehabil Eng*, vol. 29, pp. 1827–1837, 2021, epub 2021 Sep 15, PMID: 34428147, PMCID: PMC8446341.

[6] A. D. Prete, "An online method for locomotion modes recognition in wearable lower-limb exoskeletons," Laurea Magistrale in Mechanical Engineering - Ingegneria Meccanica, Politecnico di Milano, 2023, academic year: 2022-2023.

Appendices

A. Hardware of autonomyo

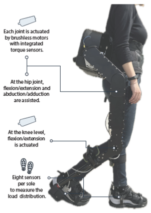
Each joint is actuated by brushless motors with integrated torque sensors.
At the hip joint, flexion/extension and abduction/adduction are assisted.
At the knee level, flexion/extension is actuated
Eight sensors per sole to measure the load distribution.

Fig. 40: Autonomyo exoskeleton

B. Path controller of autonomyo

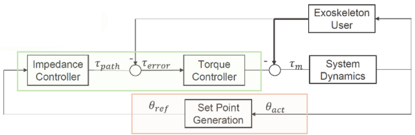

Fig. 41: Path controller of Autonomyo

C. Subjects results of final experiments

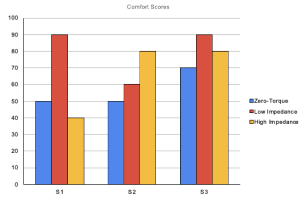

Fig. 42: Confort scores for SA experiment

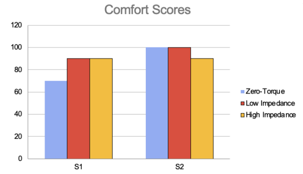

Fig. 43: Confort scores for SD experiment

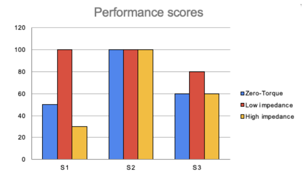

Fig. 44: Performance scores for SA experiment

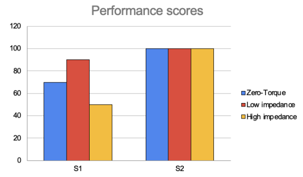

Fig. 45: Performance scores for SD experiment

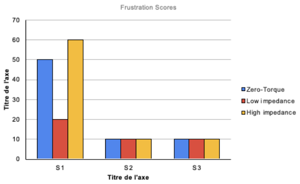

Fig. 46: Frustration scores for SA experiment

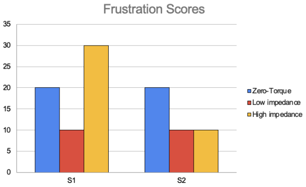

Fig. 47: Frustration scores for SD experiment

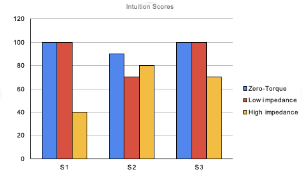

Fig. 48: Intuition scores for SA experiment

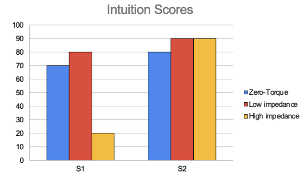

Fig. 49: Intuition scores for SD experiment

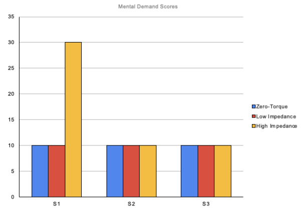

Fig. 50: Mental scores for SA experiment

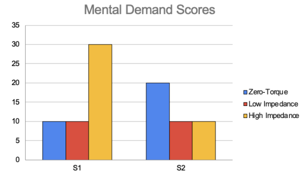

Fig. 51: Mental scores for SD experiment

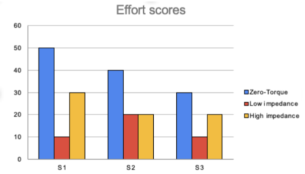

Fig. 52: Effort scores for SA experiment

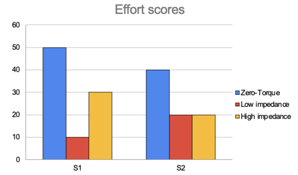

Fig. 53: Effort scores for SD experiment

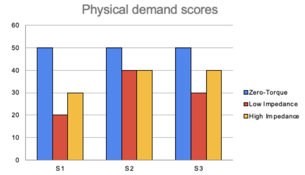

Fig. 54: Physical scores for SA experiment

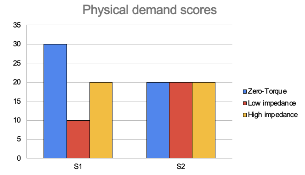

Fig. 55: Physical scores for SD experiment

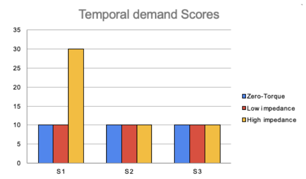

Fig. 56: Temporal scores for SA experiment

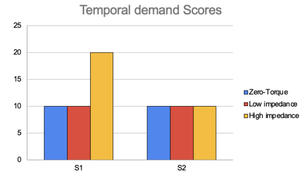

Fig. 57: Temporal scores for SD experiment