Multi-Step Planning via Signal Temporal Logic for Lower Limb Exoskeletons

1st Miroljub Mihailovic
Dept. of Information Engineering
Univ. of Padova
Padova, Italy
miroljub.mihailovic@phd.unipd.it

2nd Daniele Meli
Dept. of Computer Science
Univ. of Verona
Verona, Italy
daniele.meli@univr.it

3rd Alessandro Farinelli
Dept. of Computer Science
Univ. of Verona
Verona, Italy
alessandro.farinelli@univr.it

4th Stefano Tortora
Dept. of Information Engineering
Univ. of Padova
Padova, Italy
stefano.tortora@unipd.it

***Abstract*—Lower-limb exoskeletons (LLEs) have demonstrated considerable potential for rehabilitation, assistive, and augmentative purposes. Besides, existing devices typically lack user-friendliness and real-time adaptation to unstructured environments. We propose a modular framework for adaptive locomotion, consisting of a robotic vision module for perception of the environment and a step generation module for executing foot trajectories in real time. This research highlights the ongoing development of a multi-step planning module to generate coordinated sequences of steps that will enable the exoskeleton to run actions in advance, respecting environmental and task constraints.**

***Keywords*—Lower-limb exoskeletons, adaptive locomotion, multi-step planning, signal temporal logic (STL), neural predictive control (NPC).**

I. Introduction

Lower-limb exoskeletons (LLEs) have shown substantial potential for enhancing human mobility across rehabilitation, assistive, and augmentative applications. Despite these advances, current exoskeleton technologies still face significant challenges. Many devices lack seamless integration with the user, often resulting in discomfort and reduced usability, and they frequently fail to adapt gait patterns in real time to dynamic or unstructured environments. Addressing these limitations requires precise environmental perception and adaptive locomotion strategies capable of responding to variations in terrain and user behavior. In order to address these problems, we have developed a modular framework for adaptive locomotion. The framework currently consists of two modules. The first module is a vision module that gives the exoskeleton the ability to perceive the environment and interpret the geometry of the objects. The second module is a step generation module that can plan and execute footsteps in real time, while adapting footstep trajectories based on terrain conditions. In this work, we will introduce the current development of the framework's third module. The third module is a multi-step planning module that plans the trajectories of joint steps, allowing for the exoskeleton to plan the actions of several future steps all at once as constraints and move within the environment.

II. System Overview

The existing framework consists of two major modules: robotic vision and single-step trajectory generation. Although these modules are sufficient for reactive locomotion, they are not coordinated over multiple steps, which motivates the design of a high-level multi-step planner.

A. Robotic Vision

The robotic vision module extracts geometric features using an RGB-D camera. Multi-plane segmentation is performed with *MLESAC* [1] to detect planes such as ground, ramps, and stairs. Traversable planes are filtered by the vertical component of their normal, with the closest horizontal plane selected as the ground. The interpretation of the scene classifies environments into flat ground, staircases or ramps, and non-plane points are clustered to find potential obstacles, reducing the noise of sparse points and improving robustness for navigation and locomotion.

B. Linear Constraints Local Kernelized Movement Primitives (LCL-KMP) for Step Generation

Footstep trajectories are adapted to terrain and user-specific gait using Kernelized Movement Primitives (KMP) [2]. Swing and support trajectories are learned from demonstrations (GMM/GMR) and are locally modified to adjust the length of steps and foot placement. LCL-KMP enforces constraints such as ground contact, obstacle clearance, and joint limits that involve terrain via-points. The results showed that each of these games had smooth, feasible, and environment-aware trajectories, and were computed in real time [3].

C. Hierarchical Planning Architecture

The Figure 1 illustrates the layered planning framework. Within the architecture, the controller harnesses high-level symbolic reasoning, robotic vision, and low-level trajectory generation to address hybrid dynamics, multiphase contacts, and constraints imposed by biomechanical properties, giving rise to integrated coordinated behaviors under a changing terrain, all while adapting to the environment in real time. At the high level, Neural Predictive Control (NPC) with Signal Temporal Logic (STL) encodes timing, stability, and foot placement while maintaining feasibility through gradient-based optimization. At the low level, Kernelized Movement Primitives (KMP) generate smooth foot trajectories that adapt in real time based on desired step length and terrain geometry, naturally and feasibly responding.

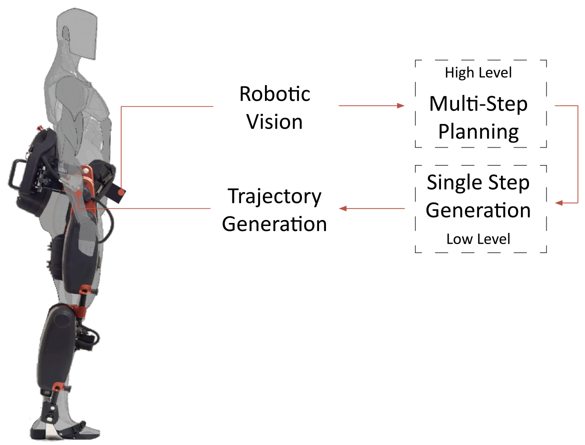

Fig. 1. System overview: the exoskeleton control framework integrates robotic vision for environment perception and multi-step planning

D. Multi-Step Planning via NPC and STL

The high-level planner generates sequences of foot placements and phase transitions using a learning-based NPC constrained by STL [4], [5]. It allows formal encoding of locomotion requirements, including balance, contact timing, and target foot placement.

Example specifications include:

- $\varphi_1 = \Box\big(p(t) \in \text{GroundRange}\big)$, ensuring that the foot remains within the ground contact region.
- $\varphi_2 = \Diamond_{[0,5]}\big(|p(t+1) - p(t)| < \varepsilon\big)$, constraining foot displacement within a 5-time-unit interval.
- $\varphi_3 = \Diamond_{[0,5]}\text{AdjustGait}(t)$, requiring gait adaptation within 5 time units.

STL-based specifications can be differentiated through robustness metrics, allowing us to perform gradient-based optimization of the neural controller [5], [6]. At each time step, the controller predicts sequences of actions for a predetermined planning horizon, assesses the sequences of trajectories produced as a function of the STL robustness score, and executes the first action of the sequence. We will also provide a backup sampling-based policy for safety in the event a specification is violated. We propose to extend the previously mentioned framework into multiple possible candidate sequences of actions per time step, acting to choose which sequence to execute based on which sequence achieved the highest STL robustness score. This mechanism allows for a more flexible approach to perception noise and reduces the potential for being trapped in poor local minima, and enhances variation in possible action sequences, thereby promoting greater generalization in unstructured environments. STL-based control has been successfully applied in robotic motion planning [7], and its compositional structure is particularly suitable for LLEs, where locomotion is decomposed into modular steps (e.g., foot swing, stabilization of contact, CoM progression) [8]. Real-time safety is further enhanced by integrating Control Barrier Functions (CBFs) [9], which can create corrective constraints anytime there are unexpected deviations.

E. Simulation Platform

The simulation environment is built in Gazebo and integrated with ROS, forming a realistic and modular testing environment to assess control strategies. The exoskeleton itself is modeled in URDF with XACRO macros to parameterize configurations for individual components such as the waist, limbs, and feet. Revolute joints actuate effort-based transmissions to provide realistic torque-controlled behavior. A Realsense D435 RGB-D camera is integrated into the system to supply perception data for vision-based algorithms. Gazebo's physics engine provides realistic simulative approaches to evaluating collisions and dynamics, while ROS manages updates for control execution and sensor data streaming. In this phase, a complete evaluation of both high-level planning and low-level trajectory generation can take place safely and in a controlled way in simulation before deployment onto a real-world physical system.

III. Conclusion

This work describes a single-step trajectory generator and visual module for adaptive human-like locomotion for lower-limb exoskeletons. The system will be adapted for multi-step planning for safer and more comfortable unsupervised use in rehabilitation; the system could also be applied to humanoid and collaborative robots. Future directions involve validation through simulation, human-in-the-loop evaluation, and implementation in real-world settings.

Acknowledgment

The authors gratefully acknowledge the support of PE00000013 "PNRR MUR - M4C2 - "Future Artificial Intelligence Research- FAIR" - SPOKE 5 - CUP B53C22003980006.

References

[1] P. Torr and A. Zisserman, "Mlesac: A new robust estimator with application to estimating image geometry," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 138–156, 2000. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314299908329

[2] Y. Huang, L. Rozo, J. Silvério, and D. G. Caldwell, "Kernelized movement primitives," *The International Journal of Robotics Research*, vol. 38, no. 7, pp. 833–852, 2019.

[3] C. Yang, X. Zhang, C. Zou, W. Liang, Z. Huang, R. Huang, Y. Wang, and H. Cheng, "Adaptive human-like gait planning for stair climbing of lower limb exoskeleton robots," in *International Conference on Intelligent Robotics and Applications*. Springer, 2024, pp. 202–221.

[4] C. Dawson and C. Fan, "Robust counterexample-guided optimization for planning from differentiable temporal logic," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 7205–7212.

[5] A. Donzé and O. Maler, "Robust satisfaction of temporal logic over real-valued signals," *Proc. Int. Conf. Formal Model. Anal. Timed Syst.*, pp. 92–106, 2010.

[6] H. A. Y.V. Pant and R. Mangharam, "Smooth operator: Control using the smooth robustness of temporal logic," in *Proc. IEEE Conf. Control Technol. Appl.*, 2017, pp. 1235–1240.

[7] H. K.-G. G. E. Fainekos, A. Girard and G. J. Pappas, "Temporal logic motion planning for dynamic robots," *Automatica*, vol. 45, no. 2, pp. 343–352, 2009.

[8] E. Plaku and S. Karaman, "Motion planning with temporal-logic specifications: Progress and challenges," *AI Commun.*, vol. 29, no. 1, pp. 151–162, 2016.

[9] L. Lindemann and D. V. Dimarogonas, "Control barrier functions for signal temporal logic tasks," *IEEE Control Syst. Lett.*, vol. 3, no. 1, pp. 96–101, 2019.