AI-Driven Human-Robot Interaction: Introducing Behaviour Trees into Smart Walkers ⋆

Paula Cardoso *, **Fabiana Machado** *, **Raquel Pereira** *, **Ricardo Mello** *, **Petter Ögren** **, **Anselmo Frizera-Neto** *

* *Electrical Engineering Department, Universidade Federal do Espírito Santo, Vitória, Brazil (e-mail: paula.a.cardoso@edu.ufes.br, ricardo.c.mello@ufes.br, anselmo.frizera-neto@ufes.br)*
** *KTH Royal Institute of Technology, Stockholm, Sweden*

Abstract: Smart walkers were developed to assist users with residual locomotion capacities providing physical support, health monitoring, and sensorial, cognitive, and navigational assistance. Multimodal human-robot interaction strategies can enable higher levels of comfort and safety when using the smart walker. Therefore, the functioning of smart walkers must be explainable and transparent, thus reflecting on the acceptance and involvement of its users. Behaviour trees are high-level language adopted in robotics due to its modularity and inherent reactivity, since its architecture facilitates the inclusion of new behaviours seamlessly.This paper proposes the use of artificial intelligence in the core of human-robot interaction strategies with focus on smart walkers to guarantee user's safety while providing locomotion assistance. The proposed approach was experimentally validated and the results indicate that the implemented behaviour tree correctly predicted unsafe scenarios whereas responding to user's commands. Such results motivate further studies upon the use of behavior trees for human-robot interaction, a research field still largely unexplored.

Keywords: human-robot interaction, behaviour tree, smart walker, assistive robot, artificial intelligence.

1. INTRODUCTION

Between the years of 2020 and 2050, 16 percent of the global population are expected to be over 65 years old, reaching a total of approximately 1.5 billion people (United Nations, 2020). Ageing is associated with the deterioration of neuromuscular and neurophysiological activities manifesting on the balance control of the elderly's gait (Osoba et al., 2019). The risk of falling is high in this population, affecting up to 85 percent of the individuals. At the age of 60 years, 85 percent of the elderly have normal gait, but this proportion is reduced to 18 percent at the age of 85 years (Sudarsky, 2001).

The human gait requires attention and functions such as the processing of internal and external information (Amboni et al., 2013). With ageing, health problems related to physical and cognitive conditioning contribute to several different gait disorders (Ghironzi et al., 2017). The fear of falling increases the elderly's attention during locomotion and, consequently, in the control of balance, which can harm and modify the biomechanics of locomotion (Sapmaz and Mujdeci, 2021). Due to the limitations of joint movements and with the decrease of muscular strength, cadence, speed and the spatio-temporal evolution of the joints differs from the normal patterns of gait (Osoba et al., 2019). Consequently, there is an interest in adopting devices that assist the mobility of people with motor disorders, offering mobility assistance and aiding in the socialization and independence of its users (Sapmaz and Mujdeci, 2021).

In this context, augmentative devices such as canes and walkers, are developed for users who have residual motor skills and are valuable options to improve the physical and cognitive state of individuals, when compared to alternative devices such as wheelchairs (Martins et al., 2012). Walkers are also potential options to rehabilitate pathological gait, and its main advantage is requiring the user's residual locomotion capacities to generate their movement, making it possible to postpone or avoid the use of wheelchairs in certain cases (Lacey and Dawson-Howe, 1997).

As an effort to leverage the benefits of conventional walkers while minimizing their downsides, robotic walkers or smart walkers (SW) have been developed by multiple research groups over the last few decades. Such devices can provide new or improved functionalities of physical support, health monitoring, and sensorial, cognitive, and navigational assistance (Martins et al., 2015). Considering the recent

⋆ This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. The authors would like to thank the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) - 304049/2019-0, 403753/2021-0, the Fundação de Amparo à Pesquisa e Inovação do Espírito Santo (FAPES) - 2021-V4J3L, 84322292, and UFES for their financial support.

integration of cloud-based functionality in such devices, SWs display several of the characteristics of other complex cyber-physical systems (CPS) (Mello et al., 2019). When making use of smart walkers, patients interact with robots directly at both physical and cognitive levels. To support locomotion, the robot must be able to infer human motion intention and respond accordingly (Cifuentes et al., 2014; Jiménez et al., 2019). Interaction strategies designed for smart walkers are usually focused on human-robot interaction (HRI) and human-robot-environment interaction (HREI) to offer features such as gait assistance, guidance, and obstacle avoidance (Jiménez et al., 2019).

Control strategies for navigation often rely on information coming from different sensors responsible for promoting safe movement (Jimenez et al., 2021). Ideally, the user should play an active role in navigating and controlling the walker, while the device works as a guide. Thus, to assure safety and proper interaction, the identification of risky situations, such as an imminent fall and excessive effort, is crucial. Due to the complexity associated with the presence of several sensors and the need for the device to react to user commands, it is pertinent to build a logic that establishes priority criterion for actions.

Nevertheless, HRI strategies presented in the literature are usually static and tailored to a certain configuration of the device. This hampers modularity, the integration of new features and the reuse of successful HRI strategies by other research groups. In this paper, we argue that leveraging artificial intelligence (AI) techniques to create high-level HRI strategies can allow for improved modularity and reactivity, which could foster and accelerate the development of complex systems.

Behaviour Trees (BTs) are designed based on goals that represents how a high-level target can be decomposed into lower levels ones in a tree structure manner (Flórez-Puga et al., 2009). Besides providing a highly modular and reactive system, BTs are also intuitive and easy to understand given its visual representation (Ögren, 2020). This can be particularly interesting in SWs, where therapists and patients should be able to understand the system and its underlying behaviour. Ultimately, if HRI is intuitive and can be understood, it can foster increased confidence in using the walker and improve the outcome of walker-aided therapies.

We propose placing AI in the core of HRI strategies, with particular focus on SWs. In this work, we discuss how BTs can be used to explore reactivity and modularity while its architecture facilitates the inclusion of new behaviours in a transparent way. This decouples SW configuration from the control strategy in action, and has the potential to enable complex HRIs that advance the state-of-the-art in SWs. We present the UFES CloudWalker, a cloud-enabled SW having multiple interaction modes, and validate the use of BTs in the system by performing a set of experiments in which the HRI is governed by a safety-focused BT.

2. BACKGROUND AND RELATED WORKS

In SWs, HRI is either based on physical or cognitive interaction - or both (Martins et al., 2012). Physical interaction happens when there is direct contact between the human and the robot, while cognitive interaction explores high-level functions performed by the human brain, such as memory and planning. Multimodal HRI strategies integrate multiple interfaces to overcome limitations and provide higher levels of comfort and safety when using SWs (Cifuentes and Frizera, 2016). Furthermore, multimodal interaction allows for exploring patterns specific to certain diseases and conditions, widening the scope of the population that can benefit from walkers. Nevertheless, the more tailored to a given use an HRI strategy is, the harder it is to leverage the strategy in other systems.

2.1 Complex HRI in SWs

SWs are devices designed to promote locomotion while performing several functionalities, such as gait assistance, partial body weight support, health monitoring and environment interaction (Martins et al., 2012). These functionalities are mostly implemented through multimodal HRI and HREI strategies. The SW presented by Wachaja et al. (2017) detects obstacles and sends haptic feedback warnings via vibration belt and motors on the handles. Alternatively, the AGoRA walker, presented by Sierra M et al. (2019), de-emphasizes users' commands once its sensors detect an obstacle, forcing the device to reduce its velocity. SWs can also be designed for a specific target population that depends on special HRI features to navigate safely. As an example, the Guido SW targets visually impaired patients (Lacey and Rodriguez-Losada, 2008), the i-Walker is adequate for elders who suffer from stroke (Morone et al., 2016) or Parkinson's disease (Ballesteros et al., 2017), and the CPWalker was developed for patients with cerebral palsy (Cifuentes et al., 2016).

To establish the supervision of the basic conditions for the safe motion of SWs, there is an interest in decision-making methods that are complex, but at the same time modular. By adopting such techniques, the robotic platform intelligently processes its own parameters and of the environment and autonomously makes decisions about its actions, improving its control capacity (Oliff et al., 2020). Currently in the SW literature there is no indication of use of a formal methodology for decision-making, as the control techniques applied tend to rely on specific configurations of the device to achieve its goals.

Robotic systems are sometimes complex and hard to understand for its users, in a way that may affect the acceptance and usage of the technology. Thus, it is particularly important for autonomous intelligent systems, such as SWs, to be explainable and transparent (De Graaf et al., 2021). Furthermore, the adoption of visual and user-friendly interfaces can simplify both task representation and execution (Han et al., 2021). Therefore, both patient and therapist can benefit from such concepts, since the risk of incorrectly representing the system is minor and the probability of patient under-trust - or over-trust - of the robot is also reduced.

2.2 BTs for Explainable Complex HRI

Behavior trees (BTs) is a hierarchical modular structure for switching between different controllers. BTs were first created by computer game programmers (Flórez-Puga et al., 2009; Isla, 2005) in an effort to increase the modularity of the AI they created using finite state machines, but are now receiving an increasing attention in the AI and robotics communities (Bagnell et al., 2012), with over 150 papers cited in recent surveys (Iovino et al., 2020). From a theoretical standpoint, BTs have been shown to be optimally modular (Biggar et al., 2020), while at the same time being intuitive for robotic end-users (Paxton et al., 2018). It is well known that modularity is a key factor in handling complexity, as it allows developers to design, test and replace different components without having to take the entire system into account. We believe that these features together is a strong motivation for exploring the use of BTs in SWs.

An example of the use of BTs in HRI can be seen in the work by Paxton et al. (2018), where users were able to create effective and perception-driven task plans for collaborative robots using the Behaviour Tree-based CoSTAR system, but many challenges still remain in this area.

Although there is no evidence of BTs being used in mobility aids or rehabilitation devices, its application in SW can be advantageous to explore reactivity and modularity, in addition to allowing the generation of behaviours to coordinate navigation according to human-robot interactions.

3. THE UFES CLOUDWALKER

The UFES CloudWalker is a cloud-enabled smart walker that possesses multiple interaction channels with the user, such as force sensors, cameras and Laser Range Finders (LRFs) (see Fig. 1) (Mello et al., 2019). This allows complex multimodal HRI and HREI. An embedded industrial computer based on the Raspberry Pi processes sensor data and manages control algorithms, whereas part of the high-level computation (e.g., machine learning algorithms and image processing) is aided by a cloud platform. To achieve the desired HRI and HREI, the control over the walker must be shared by the user and a navigation system. The overall system can be divided into six main modules, as illustrated in Fig. 2. We briefly explain the modules represented in the figure:

Human-Robot Interfaces: three interaction channels are used to extract information regarding the user's movement intentions: (i) a force feedback subsystem, based on data extracted by the force sensors mounted on the forearms supporting platforms (e.g. (Jiménez et al., 2019)); (ii) a leg-tracking subsystem, based on laser scan readings of the user's lower limbs, and; (iii) a face-tracking subsystem, to allow for steering commands based on face orientation (e.g. (Scheidegger et al., 2019)).

Movement Intent Extraction: using multiple interaction channels allows for flexible multimodal HRI techniques to accommodate patients with different impairment characteristics. This module infers the user's movement intentions according to a strategy configured by the therapist. Depending on the ideal configuration for each patient, a subset of the human-robot interfaces present in the robot are activated.

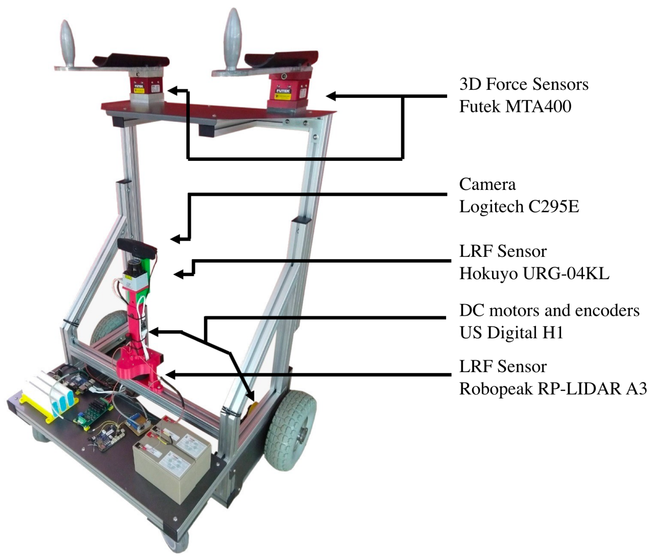

Fig. 1. The UFES CloudWalker: mechanical frame and a subset of available sensors.

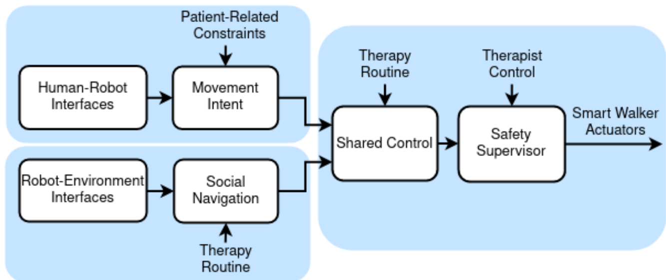

Fig. 2. General overview of the UFES CloudWalker system.

Robot-Environment Interfaces: three interfaces are used to extract information regarding the surrounding environment: (i) a localization subsystem outputs the SW's position; (ii) an obstacle detection subsystem detects the position of surrounding obstacles, and; (iii) a visual semantic subsystem outputs the pose of humans and objects.

Social Navigation: The existence (or absence) of a navigational goal is a main factor in the SW navigation. In case there is a goal position to be reached, a path must be generated; pre-programmed paths can also be set by the therapist to explore specific therapy routines (e.g., Jiménez et al. (2020)). In the absence of navigational goals, social navigation is executed to shape HRI if a person or object is detected within given proxemic zones (Jimenez et al., 2021).

Shared Control: This module integrates the information provided by the Movement Intent and Social Navigation modules to allow the user to control the SW, depending on their commands and the surrounding environment. An admittance-based controller interprets the commands as virtual forces and uses it to control the HRI and the walker displacement. The admittance controller dynamics can be tuned online by the therapist to achieve desired responses or to increase the patient's comfort.

Safety Supervisor: This module overtakes control of the SW if an unsafe situation is detected, which triggers a correction behaviour. A remote control subsystem allows the therapist to overtake control whenever necessary (e.g., to stop the walker). A fall detection subsystem identifies possible dangerous human states which may result in falls and accidents (e.g., lower limbs moving away from the walker). Finally, a collision avoidance subsystem makes sure that the SW does not move closer to obstacles than an allowed safety distance.

The integration of all these modules to generate proper HRI is complex, as there are many variables and conditions that must be observed while several computing tasks are conducted in parallel. When proposing a novel controller or HRI strategy, one may either deal with the complexities of integrating it properly with the whole system - which may not always be possible - or replace already-available features with the new feature. This limitation can be observed throughout the SW literature, affecting most - if not all - research groups. We argue that adopting modular practices enables an easier integration of innovative solutions without compromising other features, further promoting the development of SWs. Furthermore, it also enables the use of AI techniques to manage the HRI and the desired responses from the SW.

4. INTRODUCING BEHAVIOUR TREES INTO SMART WALKERS

BTs are composed by execution nodes and control nodes; signals called ticks are periodically sent from the root node down through the tree to probe nodes response - usually *success*, *failure*, or *running* (Ghzouli et al., 2020). Execution nodes can be either conditions checkers (which return *success* or *failure* after being called) or actions (which represent the actual behaviours and can be executed for longer periods of time) (Iovino et al., 2020). In diagram representation, a condition node is indicated by an oval shape whereas an action node is represented by a box; in both cases, a descriptive text lies within the symbol. Control flow nodes can be of three different categories: *Sequence*, *Fallback* and *Parallel*. *Sequences* are used when children are executed in order and are represented by an arrow inside a box. *Fallbacks* nodes are applied when children are executed in order until any child returns *success* or *running*, and are represented by a question mark. When children are executed simultaneously, the *Parallel* control flow node must be used, represented by double arrows. An example BT is shown in Fig. 3.

The BT presented in Fig. 3 comprises one condition node ("Is someone using the walker?"), and two action nodes ("Follow user's motion intention" and "Wait for user"). A *Fallback* control flow node is the root of the tree and verifies the return status of its leftmost child (the *Sequence* node). If the condition is satisfied, the action below the *Sequence* node is executed, allowing the navigation of the SW using a given HRI strategy. In case the condition fails, the *Sequence* returns the failure to the *Fallback* node, triggering the rightmost action, which ensures the SW is stationary.

Safety concerns during SW usage can either be extrinsic (e.g., colliding with an obstacle) or intrinsic (e.g., falling)

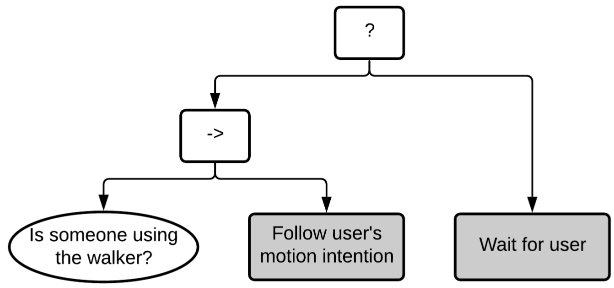

Fig. 3. Example of a simple BT diagram for a SW; the robot may only move if someone is interacting with it.

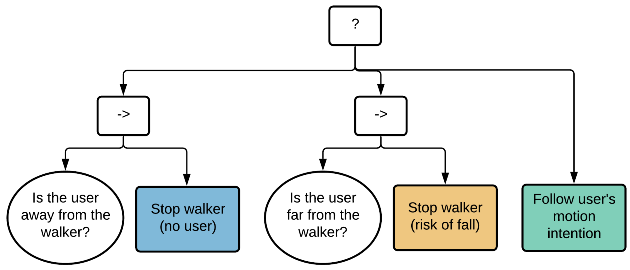

Fig. 4. HRI managed by BT in the UFES CloudWalker; safety conditions are verified to enable the use of the walker. The same colour scheme is used in Fig. 5.

to the HRI. Regarding the latter, there are two main factors that must be observed: proper use of the weight support structures and correct posture (Jiménez et al., 2019). By monitoring the interaction forces and the distance between the user's lower limbs and the structure of the walker, one might infer the stability of the user's posture and mitigate the risk of falling.

Thus, in our first implementation of BTs in the UFES CloudWalker, we designed a tree to substitute the Safety Overseer module previously used. The BT developed supervises the safety conditions to prevent fall risks and other accidents by predicting dangerous states from the HRI. Fig. 4 illustrates the implemented tree in its classical formulation; the BT verifies basic conditions to assert the presence of a user in the correct position and posture before allowing the user to move the walker. During design, we considered a backward chaining approach to prioritize safety conditions and achieve coherent global interaction considering disturbance rejection (Ögren, 2020).

As seen in Fig. 4, BTs are implemented according to the verification of two conditions regarding HRI information. Based on the values of force and distance registered during the interaction between user and walker, it is possible to predict if an unsafe condition is about to occur. By defining a threshold value and an activation zone in the interface, it is possible to identify if the user is losing physical contact with the device or if the posture is unstable, increasing the risk of falls.

In these terms, the condition *"Is the user away from the walker"* checks if there is weight-bearing, while *"Is*

ISSN: 2525-8311 DOI: 10.20906/CBA2022/3489

the user far from the walker" checks if the user's lower limbs are placed within an expected zone. Thus, the first condition verifies if the downward forces exerted upon the walker are bigger than a given threshold, as expressed by $(F_{L_Z} \leq \delta_w) \wedge (F_{R_Z} \leq \delta_w)$, where F_{L_Z} and F_{R_Z} are the downward forces measured in the left and right force sensors, respectively, and δ_w is the threshold. The second condition verifies if $(d_{ll} \leq \gamma_{min}) \wedge (d_{ll} \geq \gamma_{max})$, where d_{ll} is the average distance of the user's lower limbs from the ideal posture and γ_{min} and γ_{max} are the thresholds defining the adequate zone. Although both *"stop walker"* behaviours that result from each of these conditions merely stop the walker and are effectively the same, they were separated in the tree to make causality clear.

In case both conditions result in failure, the behaviour *"Follow user's motion intention"* is activated, and the walker responds directly to user input. Thus, the walker velocity is calculated using an admittance-based controller to allow for proper interaction. Equations 1-4 govern this controller, as suggested by Jimenez *et al.* (Jiménez et al., 2019).

$$F(t) = \frac{F_{L_Y}(t) + F_{R_Y}(t)}{2} \tag{1}$$

$$\tau(t) = \frac{(F_{L_Y}(t) - F_{R_Y}(t))d}{2} \tag{2}$$

$$v(t) = \frac{F(t) - m_v \dot{v}(t)}{b_l} \tag{3}$$

$$\omega(t) = \frac{F(t) - m_\omega \dot{\omega}(t)}{b_a} \tag{4}$$

where F_{L_Y} and F_{R_Y} are the forces applied on the Y axis of the left and right sensors (forward forces), d is the distance separating the force sensors, b_l and b_a are the linear and angular damping constants and m_v and m_ω are mass constants. This system was implemented using the Robot System Operating (ROS) and *py_trees*, a Python framework for BT development.

4.1 Validation Experiments

We validate the proposed system by verifying the correct behaviour selection and proper HRI in a set of experiments. Our objective is to demonstrate the feasibility of using BTs in SWs. We observed the interaction forces the volunteer exerted upon the walker, the distance of their lower limbs from the desired position, the controllers' response and the SW behaviour.

Four volunteers (1 female, 3 males, 28.25 ± 4) performed 5 trials each, freely conducting the SW inside a corridor in a building hall. The volunteers present no gait impairments, as our aim is observing the correct activation of the behaviours implemented in the BT. The conditions for behaviour activation were discussed with the volunteers to explain the BT. Since the volunteers were aware of the expected behaviours given the safety conditions, they were asked to navigate freely and to choose when to activate each behaviour. A trial would end after all three behaviours were observed.

4.2 Results and Discussion

All trials were successfully performed by the volunteers and each trial lasted 90 ± 10 seconds and each behaviour was activated, in average, 2 times per trial. The volunteers reported being able to understand the behaviour of the SW, as it responded directly to their actions (e.g., removing their arms from the supporting platforms or distancing themselves from the walker).

A representative result of the HRI during one of the trials is shown in Fig. 5. By the beginning of the trial, the volunteer is correctly positioned to use the walker and thus the two conditions being verified return failure, thus activating behaviour *"Follow user's motion intention"* (referred to as B3 in the figure). This can be seen in Fig. 5 during the first five seconds, where the measured weight bearing is above the threshold (second graph, top to bottom) and the distance of the user's lower limbs from the ideal position is within the desired zone (third graph, top to bottom). Thus, behaviour B3 is activated and the walker's linear velocity commands (fourth graph, top to bottom) are a direct response to the user's forward commands. Graphs for user-exerted torque and walker's angular velocity are omitted for the sake of simplicity, but the effects are similar. Behaviour activation is indicated in the bottom graph of Fig. 5 and the colours associated to a given behaviour are used as background in the other graphs.

In Fig. 5, after the mark of five seconds, the volunteer distances herself from the walker, and her lower limbs are detected outside of the preferred zone. This activates behaviour *"Stop walker (risk of fall)"* (B2 in the figure) and the walker stops moving. As soon as the volunteer approaches the walker, behaviour B3 is activated. Behaviour *"Stop walker (no user)"* is activated when incorrect conditions for weight-bearing are verified. In Fig. 5, this can be seen whenever one of the force sensors measurements indicates that there is little force being applied or even that the user is pulling the walker upwards (indicated by a negative weight). It can be observed that during the part of the trial represented in Fig. 5, the volunteer activated the behaviours to stop the walker three times.

As discussed in Section 2.2, BTs are well suited for handling hierarchical tasks. Based on the implemented design, the force-associated condition has higher priority than the other condition. When the leftmost branch of the tree in Fig. 4 results in success (*the user is away from the walker*) and the middle branch results in failure (*the user is not far from the walker*), the user is in the correct posture but not in physical contact with the walker. In order to guarantee the correct position and user safety, it is required that both arms are placed on the forearm supports. The downward force graph in Fig. 5 (third graph, top to bottom) shows the violation of this safety condition at times 12 and 21 seconds. This feature can be observed as the velocity is set to zero once there is a switch from behaviour B3 to B1. Besides this, around 12 and 15 seconds there were switches between behaviours B1 and B3, that can be avoided using control barrier function (Ames et al., 2019), as suggested by Özkahraman and Ögren (2020).

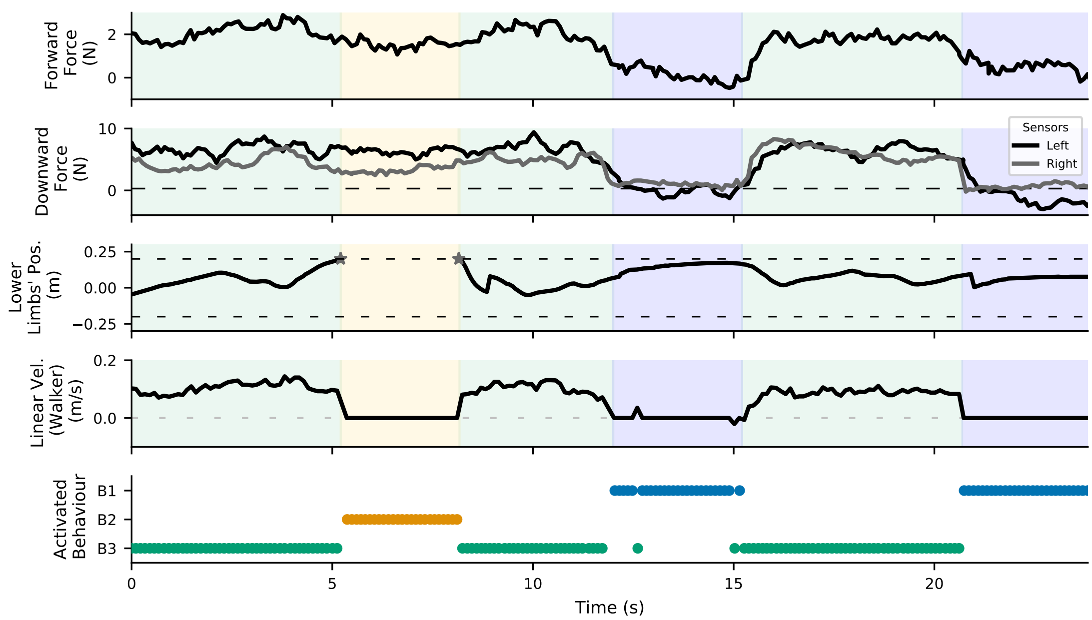

Fig. 5. Representative results of HRI from one of the trials. From top to bottom, the graphs indicate (i) the total forward force exerted by the user on the walker to control navigation; (ii) downward forces measured in each of the forearms supporting platforms; (iii) the centroid of the lower limbs position with respect to an ideal posture (markers indicate exceeded values for distance); (iv) linear velocity commands output by each behaviour internal controller, and (v) current active behaviour (B1, B2, and B3 are in agreement to the order of the behaviours presented in Fig. 4). Dashed black lines indicate associated thresholds, and background colours reinforce the indication of current behaviour.

During the experiments, the volunteers evaluated their interaction with the walker as intuitive in relation to the BT's behaviour activation and conditions verification. The user considered that the HRI guaranteed their safety when navigating with the UFES CloudWalker. The volunteers also reported that the SW reacted directly to their choice of movement, and we believe that the brief explanation of the BT design was significant to the success of the experiment. These are preliminary conclusions, and further studies are necessary to evaluate the effectiveness of the use of BTs upon user understanding of the resulting interaction.

We intend to use this first BT implementation as a basis to adapt the UFES CloudWalker's HRI and HREI strategies. We will incorporate previously developed strategies and controllers, such as the work presented by Jimenez et al. (2021), to integrate them in a single system that can be easily tailored by therapists to specific patient needs. This will enable us to introduce novel functionality into an already complex system, in an intelligent, reactive, modular and transparent way.

5. CONCLUSIONS

This paper explores the concept of AI-driven HRI by introducing BTs into SWs. We implement a safety-focused BT to allow for monitoring the state of the user during interaction. Monitored parameters are used to identify unsafe scenarios that could result in falls. Two condition checkers were added to the BT to guarantee the correct position and posture of the user. The system displayed reactive and correct response to the HRI information, stopping the walker whenever at least one of the high risk conditions were met.

The advantages of AI integration in SWs are also related to the ease of incorporating novel features into existing trees. This preliminary study points to the feasibility of employing BTs in SW systems and provides insight on how one might migrate existing functionality into a tree structure. We intend to conduct further research on AI-driven HRI for SWs, and we believe that other research groups could also benefit from such a modular and direct approach, which could facilitate the adoption of features and the re-use of code.

REFERENCES

Amboni, M., Barone, P., and Hausdorff, J.M. (2013). Cognitive contributions to gait and falls: evidence and implications. *Movement disorders*, 28(11), 1520–1533.

Ames, A.D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., and Tabuada, P. (2019). Control barrier functions: Theory and applications. In *2019 18th European Control Conference (ECC)*, 3420–3431. IEEE.

Bagnell, J.A., Cavalcanti, F., Cui, L., Galluzzo, T., Hebert, M., Kazemi, M., Klingensmith, M., Libby, J., Liu, T.Y., Pollard, N., et al. (2012). An integrated system for autonomous robotics manipulation. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2955–2962. IEEE.

Ballesteros, J., Urdiales, C., Martinez, A.B., and Tirado, M. (2017). Automatic assessment of a rollator-user's condition during rehabilitation using the i-walker platform. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(11), 2009–2017.

Biggar, O., Zamani, M., and Shames, I. (2020). On modularity in reactive control architectures, with an application to formal verification. *arXiv preprint arXiv:2008.12515*.

Cifuentes, C.A., Bayon, C., Lerma, S., Frizera, A., and Rocon, E. (2016). Human-robot interaction strategy for overground rehabilitation in patients with cerebral palsy. In *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, 729–734. IEEE.

Cifuentes, C.A. and Frizera, A. (2016). *Human-robot interaction strategies for walker-assisted locomotion*, volume 115. Springer.

Cifuentes, C.A., Rodriguez, C., Frizera-Neto, A., Bastos-Filho, T.F., and Carelli, R. (2014). Multimodal human–robot interaction for walker-assisted gait. *IEEE Systems Journal*, 10(3), 933–943.

De Graaf, M.M.A., Dragan, A., Malle, B.F., and Ziemke, T. (2021). Introduction to the special issue on explainable robotic systems. *J. Hum.-Robot Interact.*, 10(3).

Flórez-Puga, G., Gomez-Martin, M.A., Gomez-Martin, P.P., Díaz-Agudo, B., and González-Calero, P.A. (2009). Query-enabled behavior trees. *IEEE Transactions on Computational Intelligence and AI in Games*, 1(4), 298–308.

Ghironzi, G., Capeta, J.C., Cortés, A.F., and Bettelli, G. (2017). *Aging and Age-related Functional Changes*, 3–8. Cambridge University Press.

Ghzouli, R., Berger, T., Johnsen, E.B., Dragule, S., and Wasowski, A. (2020). Behavior trees in action: a study of robotics applications. In *Proceedings of the 13th International Conference on Software Language Engineering*, 196–209.

Han, Z., Giger, D., Allspaw, J., Lee, M.S., Admoni, H., and Yanco, H.A. (2021). Building the foundation of robot explanation generation using behavior trees. *ACM Transactions on Human-Robot Interaction (2021).*

Iovino, M., Scukins, E., Styrud, J., Ögren, P., and Smith, C. (2020). A survey of behavior trees in robotics and ai. *arXiv preprint arXiv:2005.05842*.

Isla, D. (2005). Handling complexity in the halo 2 ai. *Game Developers Conference*.

Jiménez, M.F., Mello, R.C., Bastos, T., and Frizera, A. (2020). Assistive locomotion device with haptic feedback for guiding visually impaired people. *Medical Engineering & Physics*, 80, 18–25.

Jiménez, M.F., Monllor, M., Frizera, A., Bastos, T., Roberti, F., and Carelli, R. (2019). Admittance controller with spatial modulation for assisted locomotion using a smart walker. *Journal of Intelligent & Robotic Systems*, 94(3), 621–637.

Jimenez, M.F., Scheidegger, W., Mello, R.C., Bastos, T., and Frizera, A. (2021). Bringing proxemics to walker-assisted gait: using admittance control with spatial modulation to navigate in confined spaces. *Personal and Ubiquitous Computing*, 1–19.

Lacey, G. and Dawson-Howe, K. (1997). Evaluation of robot mobility aid for the elderly blind. In *Proceedings of the Fifth International Symposium on Intelligent Robotic Systems*, volume 8. Citeseer.

Lacey, G.J. and Rodriguez-Losada, D. (2008). The evolution of guido. *IEEE robotics & automation magazine*, 15(4), 75–83.

Martins, M., Santos, C., Frizera, A., and Ceres, R. (2015). A review of the functionalities of smart walkers. *Medical engineering & physics*, 37(10), 917–928.

Martins, M.M., Santos, C.P., Frizera-Neto, A., and Ceres, R. (2012). Assistive mobility devices focusing on smart walkers: Classification and review. *Robotics and Autonomous Systems*, 60(4), 548–562.

Mello, R.C., Jimenez, M.F., Ribeiro, M.R.N., Laiola Guimarães, R., and Frizera-Neto, A. (2019). On human-in-the-loop cps in healthcare: A cloud-enabled mobility assistance service. *Robotica*, 37(9), 1477–1493.

Morone, G., Annicchiarico, R., Iosa, M., Federici, A., Paolucci, S., Cortés, U., and Caltagirone, C. (2016). Overground walking training with the i-walker, a robotic servo-assistive device, enhances balance in patients with subacute stroke: a randomized controlled trial. *Journal of neuroengineering and rehabilitation*, 13(1), 1–10.

Oliff, H., Liu, Y., Kumar, M., Williams, M., and Ryan, M. (2020). Reinforcement learning for facilitating human-robot-interaction in manufacturing. *Journal of Manufacturing Systems*, 56, 326–340.

Osoba, M.Y., Rao, A.K., Agrawal, S.K., and Lalwani, A.K. (2019). Balance and gait in the elderly: A contemporary review. *Laryngoscope Investigative Otolaryngology*, 4(1), 143–153.

Özkahraman, Ö. and Ögren, P. (2020). Combining control barrier functions and behavior trees for multi-agent underwater coverage missions. In *2020 59th IEEE Conference on Decision and Control (CDC)*, 5275–5282. IEEE.

Paxton, C., Jonathan, F., Hundt, A., Mutlu, B., and Hager, G.D. (2018). Evaluating methods for end-user creation of robot task plans. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 6086–6092. IEEE.

Sapmaz, M. and Mujdeci, B. (2021). The effect of fear of falling on balance and dual task performance in the elderly. *Experimental gerontology*, 147, 111250.

Scheidegger, W., De Mello, R., Sierra, S., Jimenez, M., Munera, M., Cifuentes, C., and Frizera-Neto, A. (2019). A novel multimodal cognitive interaction for walker-assisted rehabilitation therapies. In *IEEE International Conference on Rehabilitation Robotics*, volume 2019-June.

Sierra M, S.D., Garzón, M., Múnera, M., Cifuentes, C.A., et al. (2019). Human–robot–environment interaction interface for smart walker assisted gait: Agora walker. *Sensors*, 19(13), 2897.

Sudarsky, L. (2001). Gait disorders: prevalence, morbidity, and etiology. *Advances in neurology*, 87, 111–117.

United Nations (2020). World Population Ageing 2020 Highlights: Living arrangements of older persons. *World Population Ageing 2020*, 1–47.

Wachaja, A., Agarwal, P., Zink, M., Adame, M.R., Möller, K., and Burgard, W. (2017). Navigating blind people with walking impairments using a smart walker. *Autonomous Robots*, 41(3), 555–573.

Ögren, P. (2020). Convergence analysis of hybrid control systems in the form of backward chained behavior trees. *IEEE Robotics and Automation Letters*, 5(4), 6073–6080.