

Adaptive Feedback Regulator for Powered Lower-Limb Exoskeleton under Model Uncertainty

Kirtankumar J. Thakkar¹, Victor C. Paredes¹, and Ayonga Hereid¹

Abstract—This paper presents a neural network (NN) based adaptive feedback regulator to ensure the lateral and longitudinal stability and regulate the desired walking velocity of a lower-limb exoskeleton under model uncertainty. The traditional model-based controllers for lower-limb exoskeletons often fail to stabilize the robot or accurately track the desired behaviors under model uncertainties or external disturbances. This paper proposes a neural network (NN) based online adaptive regulator that compensates for the unknown changes in model parameters and external disturbances by modifying the nominal joint trajectory. A gradient descent-based delta rule is implemented to update the weights of a single layer NN, which can be efficiently performed online by design. We demonstrate the performance of the presented regulator on ATALANTE, a fully actuated lower limb exoskeleton designed for paraplegic patients. The simulation results show that the proposed approach noticeably improves stability and the tracking performance of the system, despite significant changes in model parameters and large adversarial pushes.

I. INTRODUCTION

Robotic lower-limb exoskeletons can be used to assist individuals with walking disabilities to perform normal ambulatory functions and to provide exceptional health benefits, as well as positive psychological effects ([1], [2], [3].) Many lower-limb exoskeletons heavily rely on arm-crutches and hand-tuned hierarchical control algorithms to provide supports to the user, resulting in inefficient walking and limiting the user from utilizing their upper limbs in other purposes and potentially leading to discomfort ([4], [5].) In recent years, a growing number of successful studies translate the formal model-based controllers designed for bipedal robots to powered lower-limb exoskeletons, realizing hand-free dynamic walking without the use of crutches (see [6], [7], [8], [9], [10].) By capturing the underlying dynamics more accurately, model-based approaches yield more natural dynamic walking behaviors. Nevertheless, they are very susceptible to changes in the system’s kinematic and dynamic properties and external disturbances. To realize an expected behavior, such as walking at a certain desired velocity, one needs to model the human-exoskeleton system as accurately as possible and generate an appropriate gait trajectory—typically via optimization—for this specific model. However, once the circumstance changes (e.g., the user holds heavy objects, different users, etc.), the gait generation process must be performed again to ensure the expected performance.

This work is supported by Prof. Hereid’s start-up package from the Ohio State University. The authors particularly thanks the engineers from Wandercraft for their consistent assistance in this project.

Mechanical and Aerospace Engineering,
Ohio State University, Columbus, OH, USA
{thakkar.111,paredescauna.1,hereid.1}@osu.edu

For instance, [11] showed significant velocity tracking errors when mass and inertia parameters changing by $\pm 20\%$. It is also important to note that the gait generation process is typically not fast enough to perform online due to the complex nonlinear system dynamics.

Recent studies leverage bio-inspired feedback regulators to improve the robustness of model-based controllers for bipedal robots. For instance, [12] uses the non-collocated adaptive virtual control to achieve sustained locomotion for simple bipedal robots. [13] shows that the stability of bipedal robots improves by regulating the step length of the gaits in response to pelvis velocity. [14] and [15] introduces swing foot placement regulators—which varies the swing foot landing position in response to the robot walking velocity—to stabilize 3D walking robots. While these heuristically-design regulators improve the robot’s overall stability, they still yield noticeable steady-state tracking errors of desired behaviors (e.g., forward walking velocity) in the presence of significant model and environmental uncertainties—scenarios that wearable assistive devices often have to face. While the previous methods focused on heuristic regulators, [16] developed a model-free learning-based adaptive regulator for an underactuated 3D bipedal robot, Cassie. This approach improved the steady-state tracking performance by compensating the system’s unknown dynamics with an adaptive learning scheme on the transient error dynamics of walking velocities.

Build upon our previous work in [16], this paper presents a neural network (NN) based adaptive feedback regulator for a fully actuated exoskeleton, named ATALANTE, to improve the stability and accurate tracking of desired walking velocities under model uncertainties. The whole process to realize stable walking starts from designing a nominal gait pattern via a model-based trajectory optimization. A nominal gait for lightweight biped robots generally comprises two single support domains. Keeping its heavyweight and stability in mind, we added two double support domains (see Fig. 2). In this paper, we use the Matlab package FROST ([17]) to generate the nominal gaits for a range of forward walking velocities for a specific user. A model-free neural network-based adaptive feedback regulator is then developed to regulate the walking velocity of the exoskeleton under a wide range of model uncertainties and large external disturbances. The regulator can be implemented online without any prior knowledge of the mechanical properties, neither the exoskeleton nor the patient. In particular, we developed a gradient descent delta rule to train the weights of a single layer neural network policy, making it directly transferable

arXiv:2104.11775v1 [eess.SY] 23 Apr 2021



Fig. 1: The ATALANTE exoskeleton and its kinematic tree structure, where R_b is the floating-base coordinate located at the center of the pelvis.

to robot hardware and evaluated in real-time without any modification. The primary input for the neural network is the horizontal pelvis velocity (see Section III-B) which is related to the overall stability since it is closest to the overall center of mass of the system. We are using the IMU sensor present on the pelvis link and leveraging the invariant extended Kalman filter (InEKF) ([18]) to estimate the pelvis velocity with respect to the world frame. The proposed approach shows consistent performance in simulation despite significant variation of the model properties of the system. In particular, we demonstrate its robustness for several different walking speeds of the robot for different masses, mass distributions, link geometries, and external impulsive disturbances. We further summarize the primary contributions of the present paper as follows:

- Two double support domains have been added in the design of a nominal gait, leading to an overall improvement in the stability.
- Inspired from the regulator developed for the Cassie biped robot, we have developed and implemented a NN based adaptive feedback regulator for ATALANTE.
- In the end, we have presented a study between the proposed regulator and the heuristic foot placement regulator to highlight the overall performance improvement.

The remainder of the paper is organized as follows. Section II states the system model and gait generation process of the human-exoskeleton system, Section III presents our main contribution, the NN-based adaptive feedback regulator for robust locomotion of lower-limb exoskeletons under uncertainty. Finally, Section IV shows the simulation results of the controller on ATALANTE, and concluding remarks are provided in Section V.

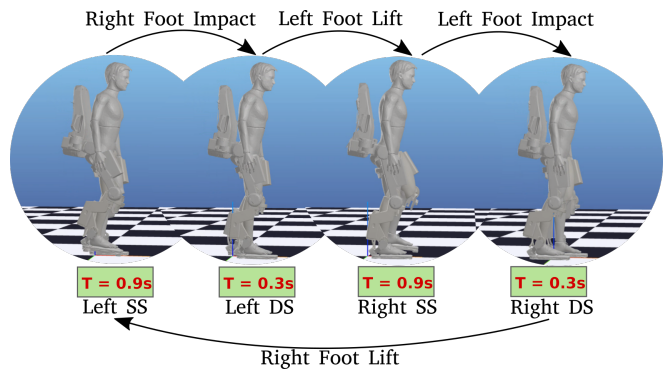


Fig. 2: A directed graph representing four domains, left single support, double support, right single support, and double support, of the periodic flat-footed walking of ATALANTE.

II. PROBLEM FORMULATION

This section, presents the mathematical formulation of the hybrid zero dynamics (HZD) based walking controller for the lower-limb exoskeleton. We consider the system dynamics's hybrid nature as it comprises a continuous walking motion and discretely modeled ground impacts. We aim to generate a continuous periodic orbit of the robot joints and enforce them via virtual constraints to realize a stable walking motion.

A. Human-Exoskeleton Lumped System Model

ATALANTE is a fully-actuated lower-limb exoskeleton designed for improving the mobility of patients with paraplegia [7]. It consists of twelve actuated joints. The weight of the device is 75 kg and can fully support up to a 90 kg user (see Fig. 1). To model the human-exoskeleton system, we consider the user body is rigidly attached to the exoskeleton to form a lumped mass system. The length of the thigh and shin links of ATALANTE can be adjusted to the user's body measurements. The mass, inertia, and center of mass of each link are approximated by the typical mass distribution of a human from its weight, assuming all links are rigid [19]. The generalized coordinates of the lumped system can be expressed by

$$q = (p, \phi, q_b) \in \mathbb{Q}, \quad (1)$$

where, $p \in \mathbb{R}^3$ and $\phi \in SO(3)$ are the global position and orientation of the floating-base coordinates rigidly attached at the pelvis link, and $q_b \in \mathbb{R}^{12}$ is the vector of relative angles of all actuated joints, as shown in Fig. 1. The walking motion of an exoskeleton is described as a hybrid control system triggered by the changes in the ground-foot contact conditions. The hybrid control system \mathcal{HC} can be described as:

$$\mathcal{HC} = \{\Gamma, \mathcal{D}, \mathcal{U}, S, \Delta, FG\}, \quad (2)$$

where, Γ is the directed graph, $\mathcal{D} \subset T\mathbb{Q}$ is a smooth manifold representing the admissible domain of the system states, \mathcal{U} represents the admissible controls, $S \subset \mathcal{D}$ is the switching surface, Δ is the reset map, and FG represents the equations of motion of the continuous dynamics, described in (3). With

respect to previous work [16], we considered a more human-like two-step periodic walking of the exoskeleton consisting of four domains, including two single domain phases and two non-instantaneous double support phases that constrains both feet to be in contact with the ground, as shown in Fig. 2. Given $u \in \mathbb{R}^{12}$ represents the actuator input vector and λ is the contact wrench vector at the contact points, the continuous dynamics of the system is given as

$$M(q)\ddot{q} + H(q, \dot{q}) = Bu + J_d^T(q)\lambda, \quad (3)$$

where $M(q)$ is the inertia matrix, $H(q, \dot{q})$ is the collection of Coriolis, centrifugal, and gravity forces, B is the actuator distribution matrix, $J_d(q)$ is the contact Jacobian matrix for a domain $d \in \mathcal{D}$. During the simple support phases, the Jacobian J_d maps the wrench from the unique contact foot. However, during double support phases, J_d maps the action of the contact wrenches on both feet by appending the appropriate individual jacobians. The discrete dynamics of the system is determined by assuming all impacts are inelastic and occur instantaneously [20]. For more details regarding the hybrid system model's construction, we refer the readers to [20], [21].

B. Gait Generation via Trajectory Optimization

Given the hybrid system defined above, we can formulate the gait generation problem as a nonlinear trajectory optimization problem using FROST. It is a Matlab based open-source software environment for developing a model-based control and motion planning for robotic systems whose dynamics is hybrid in nature. FROST formulates the trajectory optimization problem for such systems using *direct collocation* methods (see [17]). Given the hybrid system model in (2), the optimization problem can be defined as

$$\begin{aligned} & \operatorname{argmin} \sum_{d \in \mathcal{D}} L_d(q, \dot{q}, u) & (4) \\ \text{s.t. } & M(q)\ddot{q} + H(q, \dot{q}) = Bu + J_d^T(q)\lambda, & \text{(dynamics)} \\ & \delta_d(x_0, \dots, x_i, \dot{x}_i, \ddot{x}_i, \dots, \ddot{x}_N) = 0, & \text{(collocation)} \\ & (x_0^{d+}, \dot{x}_0^{d+}) = \Delta_d(x_N^d, \dot{x}_N^d), & \text{(reset map)} \\ & C_d(x_i, \dot{x}_i, u_i) \geq 0 & \text{(path constraints)} \end{aligned}$$

where, $L_d(\cdot)$ represents the cost function defined in the domain $d \in \mathcal{D}$, δ_d is the collocation constraints, Δ_d is the reset map associated with domain d and its subsequent domain d^+ , and $C_d(\cdot)$ is the collection of physical constraints.

In addition to necessary hardware constraints, such as joint motion and actuator torque limits, we employ several design-specific constraints to limit the solution space of the optimization problem to realize natural human-like walking patterns. For instance, upright torso orientation is expected during human-like walking, which can be translated into a constraint as follows,

$$\phi_{\min}^{\text{tor}} \leq \phi^{\text{tor}}(q) \leq \phi_{\max}^{\text{tor}}. \quad (5)$$

The swing foot should achieve a minimum height from the ground during a step to avoid inadvertent and early ground impacts. Similarly, higher swing foot height can be

uncomfortable for the user. This can be guaranteed by the following constraints

$$h_{\min}^{\text{sw}}(q) \leq h^{\text{sw}}(q) \leq h_{\max}^{\text{sw}}(q), \quad (6)$$

where, $h^{\text{sw}}(q)$ is the height of the swing foot, $h_{\min}^{\text{sw}}(q)$ and $h_{\max}^{\text{sw}}(q)$ are functions that profile the minimum and maximum foot clearance over a step, respectively.

Moreover, the swing foot must impact the ground with low negative velocity to produce a practical step. Note that a positive velocity would not provide a feasible solution in this context, and a large impact velocity can induce vibrations, even damages, on the robot. Such a constraint can be formulated as,

$$-\dot{h}_{\max}^{\text{sw}} \leq \dot{h}^{\text{sw}}(q_N, \dot{q}_N) \leq 0, \quad (7)$$

where, $\dot{h}_{\max}^{\text{sw}}$ is the maximum allowed impact velocity. Further, the ground reaction wrenches λ , must enforce the contact wrench cone to avoid slipping and the ZMP condition to guarantee that the support foot remains in complete contact with the ground, i.e.,

$$\lambda \in \mathcal{B}_{cws}, \quad (8)$$

where \mathcal{B}_{cws} represents the contact wrench cone determined by the geometry of the foot contacts [22].

It is important to note that the model-based gait optimization problem is constructed for a specific user's dynamical model, considering the user's geometric and weight measurements.. Each gait is also designed to achieve a specific desired walking velocity by imposing an equality constraint in the optimization.

C. Virtual Constraint based Feedback Controller

In this paper, we focus on the periodic walking of the exoskeleton at a desired forward velocity. Let π be a set of parameters that describe the desired behaviors; the virtual constraints are the difference between the actual outputs, $(\mathbf{y}^a(\mathbf{q}) \in \mathbb{R}^{12})$, and the desired outputs, $(\mathbf{y}^d(\tau, \alpha_\pi) \in \mathbb{R}^{12})$, defined as

$$\mathbf{y}(\mathbf{q}, \tau, \alpha_\pi) := \mathbf{y}^a(\mathbf{q}) - \mathbf{y}^d(\tau, \alpha_\pi), \quad (9)$$

where, $\tau = \frac{t-t_0}{T} \in [0, 1]$ is the scaled parameterization of time for each domain with t_0 being the time at the beginning of the domain, T is the duration of the domain, and \mathbf{q} is the set of actual joint angles. Here, α_π is a set of coefficients of Bézier polynomials and we are using it to represent the desired outputs here. The subscript indicates that these coefficients are determined by the desired behaviors π , typically through the gait optimization discussed in the previous subsection. The actual outputs, $\mathbf{y}^a(\mathbf{q})$, can be chosen as some kinematic functions, such as leg length or leg angle, to be directly regulated by the controller. In this paper, we particularly choose the actuated joint angles as our outputs for simplicity, i.e., $\mathbf{y}^a(\mathbf{q}) = \mathbf{q}$.

Given a set of virtual constraints, one can construct a feedback tracking controller to drive the actual outputs to the desired outputs, i.e., $\mathbf{y}(\mathbf{q}, \alpha_\pi) \rightarrow 0$. Considering the significant model uncertainty that exists in the human-exoskeleton

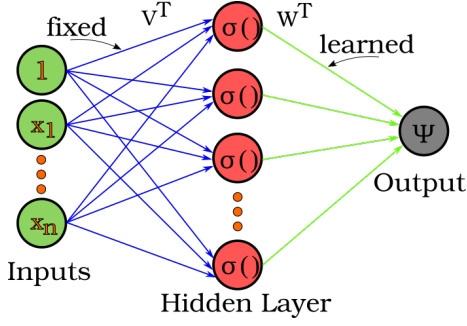


Fig. 3: The structure of a single-layer neural network policy. The weights of the first layer are randomly initialized and are kept fixed. The second layer weights are learned with the delta-rule developed in Section III.

system, we use a simple model-independent Proportional-Derivative(PD) control scheme, given as

$$u = -K_p(\mathbf{q}^a - \mathbf{q}^d) - K_d(\dot{\mathbf{q}}^a - \dot{\mathbf{q}}^d), \quad (10)$$

where K_p and K_d are the proportional and derivative control gain matrices, respectively. As discussed in Section III, the adaptability of our proposed approach will be through modifying the desired outputs, \mathbf{y}^d , via the neural-network based feedback regulator to improve the stability and behavior tracking performance.

III. NN BASED ADAPTIVE CONTROLLER

In this section, we propose an adaptive feedback regulator using a NN to realize stable walking while regulating forward walking speed in the presence of model and environmental uncertainties.

A. Review of Feedback Regulators

The presence of uncertainties in the dynamical model of a robot can generate an unstable walking motion. This leads us to the primary motivation behind the regulator design, which is essentially modifying the desired optimized trajectories to suit the real model better and enhance its stability. However, the complex dynamics of 3D bipedal robots render the regulator design improving the stability of the walking motion highly challenging. To address this challenge, researchers have started employing heuristically-designed feedback regulators (see [23], [15]) in addition to the virtual constraint based feedback controller (see Section II-C). Fig. 4 illustrates how the virtual constraint based feedback controller and the feedback regulator work in conjunction.

Foot placement is a widely used heuristically-designed regulator as highlighted in [14], [24], [15]. In particular, this regulator often uses decoupled structures, relating certain joints with specific desired features, which in this case are forward and lateral velocities. For instance, this regulator adds an offset to desired trajectories of the swing hip sagittal joint and the swing hip frontal joint (see Fig. 1) to regulate forward and lateral velocities, respectively. This modification

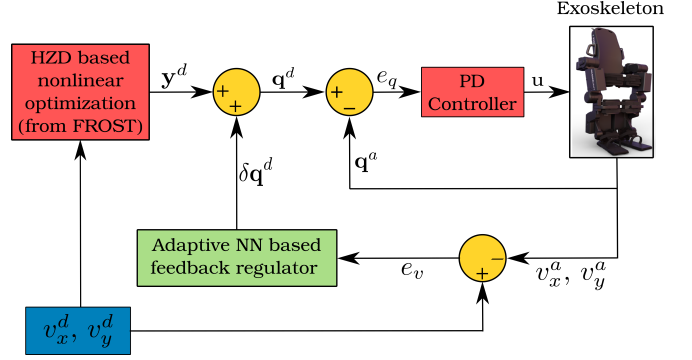


Fig. 4: The feedback structure of the adaptive regulator (green box) integrated into the virtual constraint based feedback controller.

in the desired trajectories will reshape the virtual constraints since we are choosing $\mathbf{y}^a(q) = \mathbf{q}$ as explained in the Section II-C. Modified virtual constraints can be expressed as

$$\mathbf{y}(\mathbf{q}, \tau, \alpha_\pi) := \mathbf{q}^a - (\mathbf{q}^d(\tau, \alpha_\pi) + \delta \mathbf{q}^d), \quad (11)$$

where, $\delta \mathbf{q}^d$ represents the trajectory compensation in the desired output \mathbf{y}^d , which is equal to the offset provided by a feedback regulator. Instead of producing an offset for all the desired joints angles, we focus on the most relevant ones in the regulation of longitudinal and lateral speed: swing sagittal hip and swing frontal hip, respectively, represented by $\delta \mathbf{q}_x^d[k]$ and $\delta \mathbf{q}_y^d[k]$. The formulation of this feedback regulator resembles the conventional PD controller, as expressed below.

$$\begin{aligned} \delta \mathbf{q}_x^d[k] &= K_{p_x}(v_x^d - v_x[k]) + K_{d_x}(v_x[k] - v_x[k-1]), \\ \delta \mathbf{q}_y^d[k] &= K_{p_y}(v_y^d - v_y[k]) + K_{d_y}(v_y[k] - v_y[k-1]) \end{aligned} \quad (12)$$

where, $v_x[k]$ and $v_y[k]$ are the average longitudinal and lateral speeds of the robot, v_x^d and v_y^d are the reference speeds and, $K_{p_x}, K_{d_x}, K_{p_y}, K_{d_y}$ are manually tuned gains. However, the major shortcoming of this regulator is that it can not effectively address large model uncertainties (e.g., changes in weight and its distribution) and external disturbances. For ATALANTE, the standard weight of a patient is considered to be 60kg, and the maximum allowed weight is 90kg. Considering the high level of variability, it is indispensable to have a regulator that can learn to compensate the unknown system dynamics without prior knowledge of it and address the issue of stability and forward velocity tracking. This brings us to the main contribution of this study, the adaptive feedback regulator, which is covered in detail in the following sections.

B. Adaptive Feedback Regulator

In this subsection, we propose an adaptive feedback regulator to remove the limitations of heuristically designed feedback regulators. Here, we describe the underlying idea that brought us to this specific structure of the regulator. Like a heuristically designed regulator, we aim to modify certain joint trajectories to regulate the forward and lateral speeds.

Let $v[k]$ be an average velocity at instant k and its desired value v^d (we are omitting the subscript x or y since the idea is the same for both longitudinal and lateral speed). When the robot's foot is in contact with the ground, we can express its actual velocity with the jacobian as $v[k] = J(q[k])\dot{q}[k]$. To produce a change in the velocity, we can modify the joint angles by δq . In the ideal case, with an adequate δq we aim that actual speed matches to v^d , i.e

$$v^d = J(q[k] + \delta q[k])\dot{q}[k] \quad (13)$$

$$v^d = \underbrace{J(q[k])\dot{q}[k]}_{v[k]} + \delta q[k] \frac{\partial J(q[k])}{\partial q} \dot{q}[k] + \underbrace{\mathcal{O}(\delta q, q[k], \dot{q}[k])}_{h.o.t} \quad (14)$$

$$\delta q[k] = P(\delta q, q, \dot{q})(v^d - v[k]) + Q(\delta q, q, \dot{q}) \quad (15)$$

where, P and Q are not-computed, unknown functions of $(\delta q, q, \dot{q})$. This provides the idea of adding Ψ_x and Ψ_y in our existing feedback regulators to compensate for those functions, which brings us to the foundation of the adaptive feedback regulator. Let $v_x[k]$ and $v_y[k]$ be the average longitudinal and lateral speeds of the robot in the middle of a step k , v_x^d and v_y^d be the reference speeds, we define

$$\begin{aligned} \delta \mathbf{q}_x^d[k] &= K_{p_x}(v_x^d - v_x[k]) + K_{d_x}(v_x[k] - v_x[k-1]) + \Psi_x, \\ \delta \mathbf{q}_y^d[k] &= K_{p_y}(v_y^d - v_y[k]) + K_{d_y}(v_y[k] - v_y[k-1]) + \Psi_y, \end{aligned} \quad (16)$$

where, $\delta \mathbf{q}_x^d$ will be added as an offset to desired swing sagittal hip and $\delta \mathbf{q}_y^d$ will be added to desired swing frontal hip. A detailed structure for the decoupled controllers is presented in Fig. 5.

Since our primary goal is to compensate for the unknown functions $P(\delta q, q, \dot{q})$ and $Q(\delta q, q, \dot{q})$ with added terms Ψ_x and Ψ_y , provided that the controller needs to learn them online, we refer to the widely known approaches in the neural networks community to use them as a nonlinear function approximators to learn un-modeled system dynamics. Motivated by the work of [25], [26], [27], [28], [12], [29] we use a single layer NN shown in Fig. 3 as a nonlinear function approximator without the need for computing a regression matrix. The significant advantage of NN-based controllers is that they can virtually approximate any smooth function with an appropriate number of neurons and layers.

C. Adaptive Structure

In this subsection we focus on the construction of the structure reflecting the terms Ψ_x and Ψ_y highlighted in equation (16). These are the outputs of the individual NNs multiplied with gains as expressed below:

$$\Psi_i = \mathbf{k}_i \hat{\mathbf{W}}_i^\top \sigma(\hat{\mathbf{V}}_i^\top \mathbf{x}), \quad (i = x, y) \quad (17)$$

where, \mathbf{k}_x and \mathbf{k}_y are the gains of the adaptive learning term, σ is the activation function, $\hat{\mathbf{W}}$ and $\hat{\mathbf{V}}$ are output and input layer weight matrices respectively, and \mathbf{x} is the input vector. Moreover, we have chosen the Sigmoid function as an activation function for the purpose. As shown in the Fig. 5 inputs for the NN corresponding to Ψ_x are actual

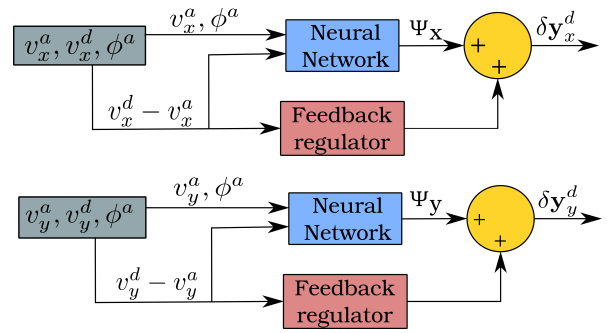


Fig. 5: Detailed structure of the proposed adaptive feedback regulator

forward velocity, torso orientation, and the error between desired and actual forward velocity. Similarly inputs for the NN corresponding to Ψ_y can be interpreted from the Fig. 5. As highlighted in [29], we are normalizing the input vector and then appending the input array by an additional input unity before feeding it to the Sigmoid function. Each of the structures has only one hidden layer with a thousand neurons.

For simplicity, we are not learning the input layer weights \hat{V} . These have been initialized with normal distribution until we found values good enough for the approximation and kept constant throughout the simulation process. \hat{W} are the output layer weights, and we are learning them via gradient descent delta-rule in an online manner. The weight update rule for the given structure can be expressed as,

$$\Delta w_{i,j} = -\gamma E_j h_i \quad (18)$$

where, $w_{i,j}$ is the weight from the i^{th} hidden neuron to the j^{th} output, E_j is the error signal for the j^{th} output, h_i is the output of the i^{th} hidden neuron, and γ is the learning rate chosen as $\gamma = 1e-4$. However, in the current scenario, we have only one output for each of the structures rendering Δw to be a vector. We have chosen the difference between desired and average actual velocity as an error signal as we aimed to regulate it.

IV. SIMULATION RESULTS

The proposed method is validated in a dynamics simulation of ATALANTE using the Jiminy Simulator (see [30]). An Invariant extended Kalman filter (InEKF) [18] has been employed to estimate the orientation of the torso and velocity of the robot from the data generated via IMU available on the torso of the exoskeleton. A single walking period consists of two single support and two double support phases. A walking gait without a double support phase could be detrimental to stability. Since its high inertia, exoskeleton, plus patient system can lead to additional external forces while switching to and from single support domains.

In this section, we present the results of the adaptive feedback regulator when the actual robot model has different mechanical properties such as mass variation and different mass distribution than the one used to generate the gait. We also present the regulator performance for different walking

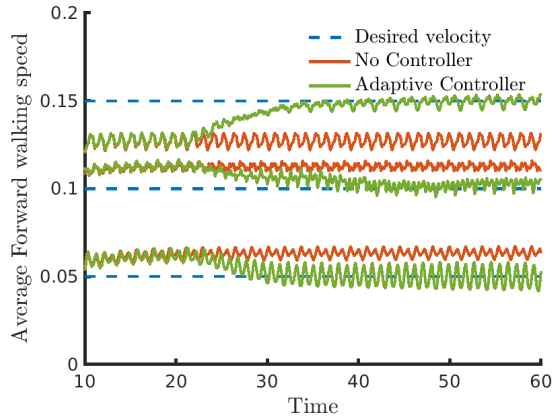


Fig. 6: Controller Performance for forward walking speed of $0.05m/s$, $0.1m/s$ and $0.15m/s$

speeds of the robot. In addition, we show the robustness of the regulator via applying external impulsive forces on the robot during the walking gait. In the end, we conclude this section by providing a comparative study between the traditional feedback regulator and the novel NN based adaptive regulator. In all the simulation (both adaptive and traditional regulator) runs, we are starting the forward foot placement regulator at 20 seconds as the robot needs around 10 seconds to reach a steady-state walking after starting from rest. However, the lateral foot placement regulator is active from the beginning of the walking motion to maintain stability in the presence of uncertainties. This also explains the slight difference in steady-state for the adaptive regulator and the conventional feedback regulator.

A. Performance for different forward speeds

First, we validated the efficacy of the regulator by tracking different forward walking speeds. Fig. 6 demonstrates the performance of the adaptive regulator for walking speeds $0.05m/s$, $0.1m/s$ and $0.15m/s$. It is interesting to observe in the same figure that the robot can not catch up with the desired forward speed when none of the controllers is employed despite the optimized trajectories are designed for that specific speed. This deviation can be attributed to the assumptions we have made while setting up the optimization problem. For instance, rigid ground impact of the foot, absence of the friction and damping in motored joints, and varied friction coefficient between ground and contact points of the foot to name a few. Moreover, it is difficult to render the error between desired and actual trajectories to absolute zero regardless of the choice of controller. This mismatch in the actual and desired trajectories can lead to a bundle of new challenges such as untimely and undesired orientation ground impact of the foot and more. These factors can work against the desired forward speed and overall stability of the robot.

Further, to highlight the regulator's capability, we present the RMS error plot (Fig. 7) of the forward velocity for $0.15m/s$. Error is evolving quite similarly until 20 seconds

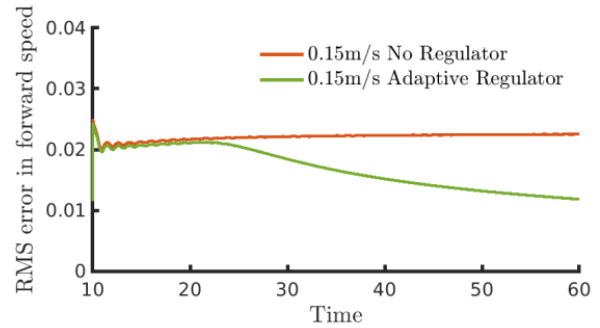


Fig. 7: RMS error in forward walking speed for $0.15m/s$

for both the cases, one without the controller and the other in the presence of the adaptive controller since the controller is inactive until then. But, after that controller takes few seconds to learn the parameters of the NN and drives RMS error plot downwards to minimize it.

B. Response to changes in the model properties of the robot

The central theme of this section is to demonstrate the efficacy of the regulator by maintaining stability and tracking forward velocity regardless of model uncertainties. For instance, variation in mass and mass distribution. We changed the model mass by varying the pelvis link (corresponding to the heaviest link) mass from its standard value in the URDF file. This is to simulate the controller performance for patients with different body weights. The gait has been generated considering a total system mass of 135kg (75kg exoskeleton mass and 60kg of mannequin mass). The maximum allowed patient mass on the exoskeleton is 90kg. Therefore, we show the regulator performance in Fig. 8 by increasing pelvis mass up to 30kg and reducing it up to 20kg.

The fundamental idea behind the powered lower-limb exoskeleton is to achieve hands-free walking. This feature can allow the user to engage the upper body in several different activities; for instance, the user carrying objects of different weights and shapes. This idea can be translated as the uncertainty in mass distribution in the simulation model. We achieve that by adding an offset to the COM location in the URDF file. We are adding an offset of a maximum $0.1m$ in forward and $0.04m$ in the backward direction. From the Fig. 9 we can verify that regulator successfully compensates for unknown dynamics and leads stable walking, ensuring convergence to desired forward velocity.

The generation of a walking gait is heavily dependent on the mechanical properties of the patient and exoskeleton system and it is evident from (4). This makes it very cumbersome to repeat the whole process for each patient with a different physical build on a larger scale. Therefore, we show the adaptability of the presented regulator by generating the walking pattern for a different model than the one in the simulation. Fig. 10 illustrates the performance of the regulator, which leads the steady velocity achieved at 20 seconds to the desired velocity of the gait.

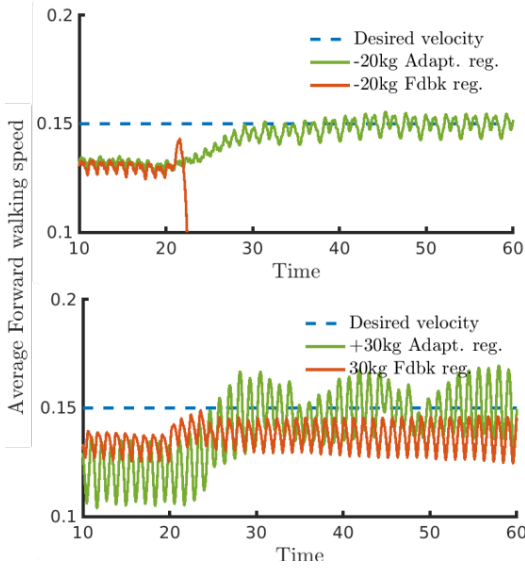


Fig. 8: Response of the novel adaptive and the conventional feedback regulator when mass of the pelvis link changed

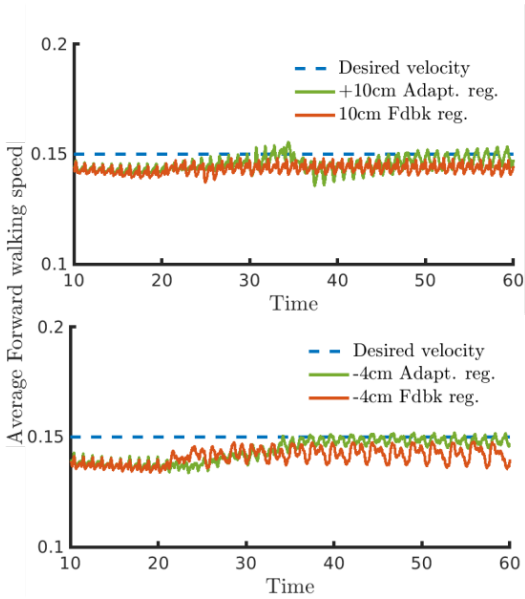


Fig. 9: Response of the novel adaptive and the conventional feedback regulator when center of mass location of the pelvis link changed in longitudinal direction

C. Robustness

We demonstrate the robustness of the regulator by validating its capability to recover the forward speed under external disturbances. For this purpose, we have applied both the forward force of $400N$ and the backward force of $200N$ for $0.1sec$ in separate simulations after $30sec$ from the start of the simulation. Fig. 11 shows that the regulator can handle external forces successfully without falling and, more importantly, recovering the tracking of the forward velocity after the disturbance is applied.

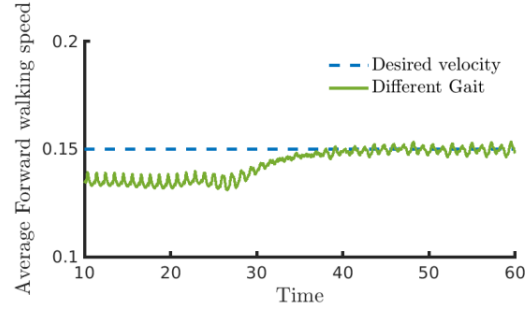


Fig. 10: Controller performance for the gait generated using different model than the one in simulation

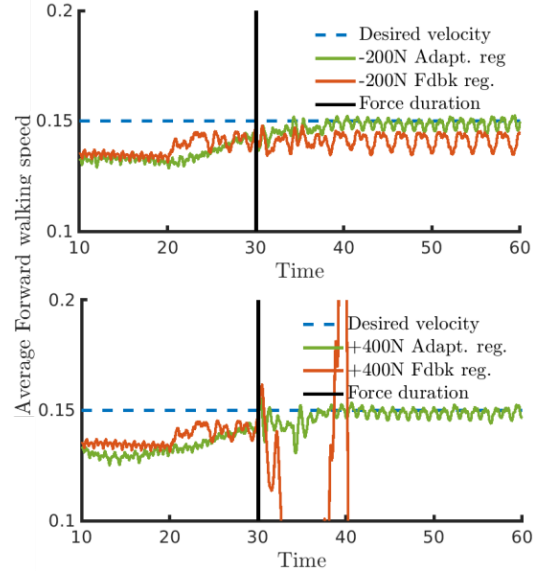


Fig. 11: Performance of the novel adaptive and the conventional feedback regulator when forward and backward impulsive forces are applied

D. Comparison between Heuristic and Adaptive controller

To illustrate the impact of the adaptive regulator, we are also presenting a comparative study against the traditional feedback regulator. Fig. 8 shows the forward velocity tracking performance of both the regulators for a walking speed of $0.15m/s$. When the pelvis link's mass was reduced by $20kg$ traditional feedback regulator failed to maintain the stability, let alone the tracking performance. Similar behavior can be observed for the feedback regulator from the Fig. 11 when $200N$ of impulsive force applied in the backward direction. Although, for the traditional regulator case robot did not fall when the location of COM changed (see Fig. 9) but walking motion develops a steady state error in velocity tracking. In summary, either the traditional regulator fails to balance the robot or develops steady state tracking error when pushed to the limits. On the contrary, the novel adaptive feedback regulator shows consistent performance regardless of the uncertainty put forth.

V. CONCLUSION

In this paper, we propose the addition of two double support domains in a nominal walking gait. We also present the design of the adaptive regulator leveraging the idea of NN for nonlinear function approximation and evaluate its performance over the ATALANTE platform. The regulator assures the tracking of the forward velocity of an exoskeleton under the presence of model and environmental uncertainties. The performance has been demonstrated using the Jiminy simulator. The presented regulator builds upon the existing ideas of the feedback regulator to handle the uncertainties more effectively. It gradually learns and converges at the modifications for the specific joint trajectories. The regulator's online learning nature makes it even more effective to learn unknown system dynamics without any prior training. This regulator has excellent performance on the simulator. However, it lacks the experimental performance evaluation, which is the next target for this research. Moreover, the regulator has been evaluated particularly for an exoskeleton but being able to learn system dynamics, it can be implemented for any bipedal walking robot for stability and forward velocity tracking.

REFERENCES

- [1] K. Raab, K. Krakow, F. Tripp, and M. Jung, "Effects of training with the ReWalk exoskeleton on quality of life in incomplete spinal cord injury: a single case study," *Spinal cord series and cases*, vol. 2, no. 1, pp. 1–3, 2016.
- [2] P. R. Geigle and M. Kallins, "Exoskeleton-assisted walking for people with spinal cord injury," *Archives of physical medicine and rehabilitation*, vol. 98, no. 7, pp. 1493–1495, 2017.
- [3] H. I. Krebs, N. Hogan, M. L. Aisen, and B. T. Volpe, "Robot-aided neurorehabilitation," *IEEE transactions on rehabilitation engineering*, vol. 6, no. 1, pp. 75–87, 1998.
- [4] R. Jimenez-Fabian and O. Verlinden, "Review of control algorithms for robotic ankle systems in lower-limb orthoses, prostheses, and exoskeletons," *Med. Eng. Phys.*, vol. 34, no. 4, pp. 397–408, 2012.
- [5] K. Anam and A. A. Al-Jumaily, "Active exoskeleton control systems: State of the art," *Journal of Procedia Engineering*, vol. 41, pp. 988–994, 2012.
- [6] A. Agrawal, O. Harib, A. Hereid, S. Finet, M. Masselin, L. Praly, A. D. Ames, K. Sreenath, and J. W. Grizzle, "First steps towards translating HZD control of bipedal robots to decentralized control of exoskeletons," *IEEE Access*, vol. 5, pp. 9919–9934, 2017.
- [7] O. Harib, A. Hereid, A. Agrawal, T. Gurriet, S. Finet, G. Boeris, A. Duburcq, M. E. Mungai, M. Masselin, A. D. Ames, K. Sreenath, and J. Grizzle, "Feedback control of an exoskeleton for paraplegics: toward robustly stable hands-free dynamic walking," *IEEE Control System Magazine*, vol. 38, no. 6, pp. 61–87, Dec. 2018.
- [8] T. Gurriet, S. Finet, G. Boeris, A. Duburcq, A. Hereid, O. Harib, M. Masselin, J. Grizzle, and A. D. Ames, "Towards restoring locomotion for paraplegics: realizing dynamically stable walking on exoskeletons," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 2804–2811.
- [9] A. Duburcq, Y. Chevaleyre, N. Bredeche, and G. Boeris, "Online trajectory planning through combined trajectory optimization and function approximation: Application to the exoskeleton atalante," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 3756–3762.
- [10] M. Tucker, E. Novoseller, C. Kann, Y. Sui, Y. Yue, J. W. Burdick, and A. D. Ames, "Preference-based learning for exoskeleton gait optimization," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2351–2357.
- [11] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. R. Westervelt, C. C. De-Wit, and J. W. Grizzle, "RABBIT: a testbed for advanced control theory," *IEEE Control Systems*, vol. 23, no. 5, pp. 57–79, Oct. 2003.
- [12] M. Gnucci and R. Marino, "On the adaptive control of the acrobot," in *2019 18th European Control Conference (ECC)*, June 2019, pp. 3438–3443.
- [13] Y. Aoustin and A. Formal'sky, "Control design for a biped: Reference trajectory based on driven angles as functions of the undriven angle," *International Journal of Journal of Computer and Systems Sciences*, vol. 42, no. 4, pp. 645–662, 2003.
- [14] X. Da, O. Harib, R. Hartley, B. Griffin, and J. W. Grizzle, "From 2d design of underactuated bipedal gaits to 3d implementation: Walking with speed tracking," *IEEE Access*, vol. 4, pp. 3469–3478, 2016.
- [15] Y. Gong, R. Hartley, X. Da, A. Hereid, O. Harib, J.-K. Huang, and J. Grizzle, "Feedback control of a Cassie bipedal robot: walking, standing, and riding a segway," *American Control Conference (ACC)*, 2019.
- [16] G. A. Castillo, B. Weng, T. C. Stewart, W. Zhang, and A. Hereid, "Velocity regulation of 3d bipedal walking robots with uncertain dynamics through adaptive neural network controller," 2020.
- [17] A. Hereid and A. D. Ames, "Frost: Fast robot optimization and simulation toolkit," pp. 719–726, 2017.
- [18] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended kalman filtering for robot state estimation," *The International Journal of Robotics Research*, vol. 39, no. 4, pp. 402–430, 2020. [Online]. Available: <https://doi.org/10.1177/0278364919894385>
- [19] D. A. Winter, *Biomechanics and motor control of human movement*. John Wiley & Sons, 2009. [Online]. Available: <https://onlinelibrary-wiley-com.proxy.lib.ohio-state.edu/doi/book/10.1002/9780470549148>
- [20] J. W. Grizzle, C. Chevallereau, R. W. Sinnet, and A. D. Ames, "Models, feedback control, and open problems of 3D bipedal robotic walking," *Automatica*, vol. 50, no. 8, pp. 1955–1988, 2014.
- [21] A. Hereid, C. M. Hubicki, E. A. Cousineau, and A. D. Ames, "Dynamic humanoid locomotion: a scalable formulation for HZD gait optimization," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 370–387, Apr. 2018.
- [22] H. Dai, A. Valenzuela, and R. Tedrake, "Whole-body motion planning with centroidal dynamics and full kinematics," in *2014 14th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*. IEEE, 2014, pp. 295–302.
- [23] S. Kolathaya, W. Ma, and A. D. Ames, "Composing dynamical systems to realize dynamic robotic dancing," in *Algorithmic Foundations of Robotics XI - Selected Contributions of the Eleventh International Workshop on the Algorithmic Foundations of Robotics, WAFR 2014, 3-5 August 2014, Boğaziçi University, İstanbul, Turkey*, ser. Springer Tracts in Advanced Robotics, H. L. Akin, N. M. Amato, V. Isler, and A. F. van der Stappen, Eds., vol. 107. Springer, 2014, pp. 425–442. [Online]. Available: https://doi.org/10.1007/978-3-319-16595-0_25
- [24] S. Rezazadeh, C. Hubicki, M. Jones, A. Peekema, J. Van Why, A. Abate, and J. Hurst, "Spring-mass walking with atrias in 3d: robust gait control spanning zero to 4.3 kph on a heavily underactuated bipedal robot," in *ASME 2015 Dynamic Systems and Control Conference*. American Society of Mechanical Engineers, 2015.
- [25] K. S. Narendra and S. Mukhopadhyay, "Adaptive control using neural networks and approximate models," *IEEE Transactions on Neural Networks*, vol. 8, no. 3, pp. 475–485, May 1997.
- [26] R. M. Sanner and J. E. Slotine, "Stable adaptive control of robot manipulators using "neural" networks," *Neural Computation*, vol. 7, no. 4, pp. 753–790, July 1995.
- [27] F. Chen and H. Khalil, "Adaptive control of nonlinear systems using neural networks," *International Journal of Control*, vol. 55, no. 6, pp. 1299–1317, 1992.
- [28] T. DeWolf, T. C. Stewart, J.-J. Slotine, and C. Eliasmith, "A spiking neural model of adaptive arm control," *Proceedings of the Royal Society B*, vol. 283, no. 48, 2016.
- [29] F. L. Lewis, A. Yesildirak, and S. Jagannathan, *Neural Network Control of Robot Manipulators and Nonlinear Systems*. USA: Taylor and Francis, Inc., 1998.
- [30] A. Duburcq, "Jiminy," <https://github.com/Wandercraft/jiminy>, 2019, accessed: 2021-01-07.