Novel solutions on model-based and model-free robotic-assisted ankle rehabilitation

Juan Carlos Arceo Luzanilla, Jorge Alvarez, Carlos Daniel Armenta Moreno, Jimmy Lauber, Sylvain Cremoux, Emilie Simoneau-Buessinger, Miguel Bernal

10.24425/acs.2021.136878

Archives of Control Sciences
Volume 31(LXVII), 2021
No. 1, *pages* 5–27

Novel solutions on model-based and model-free robotic-assisted ankle rehabilitation

Juan Carlos ARCEO, Jorge ÁLVAREZ, Carlos ARMENTA, Jimmy LAUBER, Sylvain CREMOUX, Emilie SIMONEAU-BUESSINGER and Miguel BERNAL

In this report, ankle rehabilitation routines currently approved by physicians are implemented via novel control algorithms on a recently appeared robotic device known as the motoBOTTE. The physician specifications for gait cycles are translated into robotic trajectories whose tracking is performed twofold depending on the availability of a model: (1) if obtained via the Euler-Lagrange approach along with identification of unknown plant parameters, a new computed-torque control law is proposed; it takes into account the parallel-robot characteristics; (2) if not available, a variation of the active disturbance rejection control technique whose parameters need to be tuned, is employed. A detailed discussion on the advantages and disadvantages of the model-based and model-free results, from the continuous-time simulation to the discrete-time implementation, is included.

Key words: active disturbance rejection, computed torque control, differential algebraic equations, parallel rehabilitation robot, real-time implementation, system identification

1. Introduction

Mainly as a result of a stroke, a mild form of paralysis known as paretic, causing functional dependence, is growing all around the world; medical care,

J.C. Arceo (corresponding author, e-mail: juancarlos.arceo@uphf.fr), C. Armenta, J. Lauber, and E. Simoneau-Buessinger are with Université Polytechnique Hauts-de-France, LAMIH UMR CNRS 8201, F-59313 Valenciennes, France.

J. Álvarez and M. Bernal are with Sonora Institute of Technology, 5 de Febrero 818 Sur, Ciudad Obregon, Sonora, Mexico.

S. Cremoux is with Centre de Recherche Cerveau et Cognition, CNRS UMR 5549, Université de Toulouse, Toulouse 31052, France.

This work has been supported by the ECOS Nord SEP-CONACYT-ANUIES Project (France M17M08 / Mexico 291309). This research is sponsored by ELSAT 2020 of the Hauts-de-France Region, the European Community, the Regional Delegation for Research and Technology, the French Ministry of Higher Education and Research, and the French National Center for Scientific Research.

Received 3.06.2020. Revised 15.12.2020.

unfortunately, is focused on patients with serious impairment, disregarding those who may improve significantly if assisted [6]. Due to the augmentation of the life expectancy, the latter group is steadily growing [14, 27], which combined with the increasing costs of health care following a stroke, which are among the highest [8], makes the need of technological solutions an important subject. In recent years, rehabilitation programs to recover motor functions have increasingly involved a variety of disciplines beyond the traditional human-human approach. One of the most promising techniques to reduce the motor dependence of stroke survivors involves the training of the affected limbs by rehabilitation robots, which can be adapted to the level of impairment and complement the therapist indications.

The goal for robot-based rehabilitation exercises is to promote motor recovery of the affected limbs by performing repetitive tasks [12]. To attain this goal, the robot is usually a wearable device, which thus become a sort of exoskeleton; see, for instance, [18, 39] for lower-limb examples as those considered in this work. In [5], an exoskeleton device has been successfully employed for an active ankle support, where the patient is supposed to be able to walk; this is not the case in most of stroke-hit patients for which robot assistants are usually voluminous and expensive [20]. Moreover, constructing a device for assisted rehabilitation requires controller design in order to reproduce clinically validated routines; this task has been tackled with a variety of control techniques for trajectory tracking of joint positions and velocities, for instance: proportional [34], feedforward [43,54], and torque control [31]. Importantly, the latter technique is only adequate if the device is an open kinematic chain.

In this context, the Laboratory of Industrial and Human Automation Control, Mechanical Engineering and Computer Science (LAMIH) developed a transportable robot to perform ankle rehabilitation routines on stroke-hit patients [24]. The motoBOTTE (see Fig. 1) is a one-degree-of-freedom parallel robot to be used along constant monitoring of the patient performance; the sensor and control signals are computer-processed. As shown in [3], such device can significantly reduce the therapy costs while providing a greater flexibility and adaptation of the required routines to the corresponding case.

A parallel robot, such as the motoBOTTE, is a structurally closed kinematic chain mechanism [28] that is capable of moving large loads with high precision [11]. The geometrical structure of such parallel robots imposes constraints on their position, speed, acceleration, and degrees-of-freedom [49]; their trajectories are restricted to a manifold that satisfies all these constraints at once [37]. Parallel robots dynamics can be described using differential algebraic equations (DAEs), also known as singular systems, semi-state representations, implicit differential equations, among other names, which arise in several areas, not only in robotics [25]. Controller design for DAEs is a challenging task for many reasons: the state space representation is not an ordinary one, but a descriptor [4];

Figure 1: The motoBOTTE

algebraic restrictions oblige to consistent initialization and holding [32]; numerical simulation cannot be directly performed via ordinary differential equations (ODEs) [30].

The modelling of parallel robots can be done combining the Euler-Lagrange approach (which is commonly used for modelling serial robots [11]) with a differential index reduction procedure [32, 37] to find passive joints dynamics. Once a mathematical description of the system has been obtained, one of the many model-based strategies for trajectory tracking can be applied: PD-control [10], sliding modes [36], fuzzy logic control [26], backstepping [50], computed torque control [42]. In this work, the latter is enhanced as to fit the DAE framework; a preliminary result can be found in [2].

Computed torque requires exact knowledge of the system model and parameters, otherwise the control laws might be ineffective [11]. Our second contribution addresses the case where a model is not available: we develop and implement a model-free methodology derived from active disturbance rejection control (ADRC), a recent technique that claims to be an upgrade on proportional-integral-derivative (PID) controllers [17]; its adaptation to parallel robots proves to be useful despite the lack of mathematical guarantees; former works on PID control of parallel robots are [22, 47].

This paper is organized as follows: in section 2 the motoBOTTE is modelled as a set of DAEs, which reflects its parallel characteristics in the form of algebraic restrictions; since the model parameters are unknown, the identification process is described; section 3 develops a novel computed-torque control scheme for parallel robots, which is applied in simulation to the motoBOTTE model of the previous section; for the case where a model is not available, section 4 extends the ADRC

methodology to parallel robots and illustrates its effectiveness on the motoBOTTE model; section 5 describes the real-time implementation of the model-free and model-based control laws developed in the previous sections; it includes details on hardware setup, discretization, unmeasured variables, controller tuning, and results on clinically approved rehabilitation routines; an ample discussion on the methodologies proposed is provided in section 6; finally, in section 7 conclusions based on the results and some insights about future work are provided.

2. Modelling of the motoBOTTE as a DAE

As stated before, computed-torque control critically depends on the model precision; thus, the motoBOTTE model is developed here. Nevertheless, keep in mind that even for ADRC which does not require the mathematical model of the plant, it is important to understand the underlying dynamics so judicious tuning and insights can be made. The motoBOTTE, shown schematically in Fig. 2, is actuated by a linear piston whose variable length is denoted by d with an operation range $\Omega = d \in [0, 0.103]$ in meters, the ankle angle ϕ is measured via the voltage of a potentiometer in the piston [3,46]. The dynamic of the ideal noise-free piston is described by

$$\dot{d} = b_1 u, \tag{1}$$

where u is the control signal, b_1 is a constant term that will be estimated in subsection 2.3 and there is an input saturator such that $u \in [-10\text{ V}, 10\text{ V}]$. Previously, a model for the motoBOTTE was identified using a black box model approach, but it does not provide information about how forces interact in the system [3]. The following procedure will take into account these forces; a preliminary work on this direction can be found in [2].

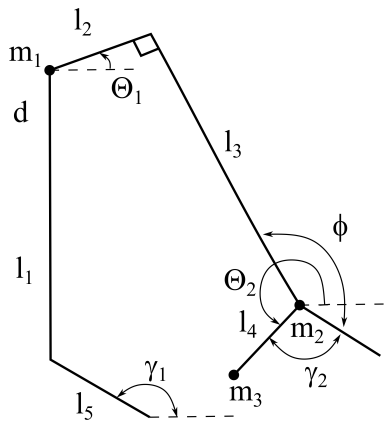


Figure 2: Scheme of the motoBOTTE

The steps for modelling a non-redundant parallel robot can be summarized as:

1. Open the kinematic chain and follow the Euler-Lagrange methodology to obtain the actuated or open-loop dynamics of the system.
2. The dynamics for the passive joints can be obtained by substituting the actuated dynamics into the second-time derivative of the algebraic constraints.

This procedure is illustrated in the next two subsections.

2.1. Open-Loop kinematics

Consider the system in Fig. 2 with punctual masses m_1, m_2, m_3, the length of the links l_1, l_2, l_3, l_4, l_5, the angles between the links and the horizontal plane Θ_1, Θ_2, γ_1, and the fixed inner angle γ_2. The total kinetic energy is given by $K = K_1 + K_2 + K_3$ with

$$K_1 = \frac{1}{2} m_1 \dot{d}^2, \qquad K_2 = \frac{1}{2} m_2 \left(\left(\dot{d} + z_1 \dot{\Theta}_1 \right)^2 + z_3^2 \dot{\Theta}_1^2 \right),$$
$$K_3 = \frac{1}{2} m_3 \left(\left(\dot{d} + z_1 \dot{\Theta}_1 + z_2 \dot{\Theta}_2 \right)^2 + \left(z_3 \dot{\Theta}_1 + z_4 \dot{\Theta}_2 \right)^2 \right),$$

as the kinetic energies of individual joints. The potential energy is $P = P_1 + P_2 + P_3$ with

$$P_1 = (l_1 + d + z_5)\, m_1 g,$$
$$P_2 = (l_1 + d - z_3 + z_5)\, m_2 g,$$
$$P_3 = (l_1 + d - z_3 + z_5 - z_4)\, m_3 g,$$

as individual potential energies with $z_1 = l_2 \cos\Theta_1 + l_3 \sin\Theta_1$, $z_2 = l_4 \cos\Theta_2$, $z_3 = l_3 \cos\Theta_1 - l_2 \sin\Theta_1$, $z_4 = -l_4 \sin\Theta_2$, $z_5 = l_5 \sin\gamma_1$, and $g = 9.806$.

Since the Lagrange equation for a conservative system [11] is $L = K - P$, the dynamical equations for the open-loop kinematics are $\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} = \tau$, where q is a n-vector of generalized coordinates composed by the prismatic joint d and two revolute joint angles Θ_1 and Θ_2 in radians; this yields $q = \begin{bmatrix} d & \Theta_1 & \Theta_2 \end{bmatrix}^T$ with τ as a vector of generalized external forces. The Euler-Lagrange equations can be written in the matrix form:

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) + F(\dot{q}) = \tau, \tag{2}$$

whose matrices are defined as:

$$C\left(\dot{q}, q\right) = \begin{bmatrix} (m_2 + m_3) z_3 \dot{\Theta}_1^2 + m_3 z_4 \dot{\Theta}_2^2 \\ m_3 \left(z_1 z_4 - z_2 z_3\right) \dot{\Theta}_2^2 \\ m_3 \left(z_2 z_3 - z_1 z_4\right) \dot{\Theta}_1^2 \end{bmatrix}, \qquad \tau = \begin{bmatrix} b_0 \dot{u} \\ 0 \\ 0 \end{bmatrix},$$

$$M(q) = \begin{bmatrix} m_1 + m_2 + m_3 & (m_2 + m_3)\, z_1 & m_3 l_4 z_2 \\ (m_2 + m_3)\, z_1 & (m_2 + m_3)\left(l_2^2 + l_3^2\right) & M_{23} \\ m_3 l_4 z_2 & M_{23} & m_3 l_4^2 \end{bmatrix},$$

with $M_{23} = m_3 l_4 (z_1 z_2 + z_3 z_4)$, b_0 is an unknown parameter that will be estimated in subsection 2.3 and $\dot{u}$ corresponds to the time-derivative of the control signal. Gravity does not affect the system due to the piston mechanical structure [46] and the configuration of the motor driver [29], therefore, it will be considered as $G(q) = 0$ and be omitted in developments thereafter. The measurable parameters of the system are presented in Table 1.

Table 1: Measured parameters of the system

Param.	Value	Param.	Value
l_1	0.35 m	l_2	0.125 m
l_3	0.445 m	l_4	0.1014 m
l_5	0.15 m	m_1	6.1514 kg
m_2	2.1398 kg	m_3	2.8684 kg
γ_1	2.4086 rad	γ_2	2.0952 rad

Assuming that friction is a local effect [11] with unknown dynamics that can be described by a continuously differentiable function within a region of interest Ω (as in the viscous friction case), there exists, by Taylor-series, a polynomial that can approximate it [40] as

$$F\left(\dot{q}\right) = \begin{bmatrix} v_{11}\dot{d}^2 + v_{12}\dot{d} \\ v_{21}\dot{\Theta}_1^2 + v_{22}\dot{\Theta}_1 \\ v_{31}\dot{\Theta}_2^2 + v_{32}\dot{\Theta}_2 \end{bmatrix},$$

where the polynomial coefficients v_{11}, v_{12}, v_{21}, v_{22}, v_{31} and v_{32} are unknown and will be estimated in subsection 2.3.

2.2. Dynamical equations of the passive joints

Once the open-loop kinematics are closed to yield the motoBOTTE structure, the vector of generalized coordinates q must satisfy the following set of algebraic constraints inherited from its geometric structure [49]:

$$\begin{aligned} d + l_1 + l_2 \sin\Theta_1 - l_3 \cos\Theta_1 + l_4 \sin\Theta_2 + l_5 \sin\gamma_1 &= 0, \\ l_2 \cos\Theta_1 + l_3 \sin\Theta_1 + l_4 \cos\Theta_2 + l_5 \cos\gamma_1 &= 0, \\ \phi = \Theta_1 - \Theta_2 + \pi/2 - \gamma_2, & \end{aligned} \tag{3}$$

where the first two restrictions arise from $x-$ and $y-$axis oriented requirements for closure and the third one is a constraint for the ankle angle. These restrictions can be combined to determine the vector of generalized coordinates q using the measured variable ϕ, which yields

$$\begin{aligned} \Theta_2 &= \arcsin\left(\frac{-l_2\cos\gamma_1}{\sqrt{\alpha_1^2+\alpha_2^2}}\right) - \arctan\left(\frac{\alpha_1}{-\alpha_2}\right), \\ \Theta_1 &= \phi + \gamma_2 - \pi + \Theta_2\,, \qquad d = -l_1 + z_3 + z_4 + z_5\,, \end{aligned} \tag{4}$$

with $\alpha_1 = l_2\cos\left(\phi+\gamma_2-\frac{\pi}{2}\right)+l_3\cos\left(\phi+\gamma_2-\pi\right)+l_4$ and $\alpha_2 = l_2\sin\left(\phi+\gamma_2-\frac{\pi}{2}\right)+l_3\sin\left(\phi+\gamma_2-\pi\right)$. The constraints (3) also impose others in the trajectory derivatives $\dot{q}$ and $\ddot{q}$; for the first-order derivative case these are

$$\begin{aligned} \dot{d} + z_1\dot{\Theta}_1 + z_2\dot{\Theta}_2 &= 0, \\ z_3\dot{\Theta}_1 + z_4\dot{\Theta}_2 &= 0, \end{aligned} \tag{5}$$

while for the 2nd-order derivative the constraints obtained are

$$\begin{aligned} \ddot{d} + z_1\ddot{\Theta}_1 + z_2\ddot{\Theta}_2 + z_3\dot{\Theta}_1^2 + z_4\dot{\Theta}_2^2 &= 0, \\ z_3\ddot{\Theta}_1 + z_4\ddot{\Theta}_2 - z_1\dot{\Theta}_1^2 - z_2\dot{\Theta}_2^2 &= 0. \end{aligned} \tag{6}$$

Now we have expressions for describing the dynamics of the passive joints.

Remark 1 *The dynamics of the motoBOTTE are obtained by selecting the actuated dynamics $\ddot{d}$ in (2) and substituting it in (6), which concludes this two-step modelling method. Notice that this is a single model for the parallel robot, the model itself is a combination of two set of equations, the first equations (2) provide a description for the behavior of the actuator and the equations in (6) describe the dynamics that the passive joints must have to satisfy the algebraic constraints inherit to the system. Therefore, our model is a set of differential-algebraic-equations (DAEs).*

The procedure above, i.e., obtaining explicit expressions for 'missing' or 'implicit' dynamics for a set of DAEs is known as the Pantelides algorithm [32], which has been already implemented in the Symbolic Math Toolbox™ in MATLAB ® [16, 41]. When this procedure is combined with the Euler-Lagrange approach as in this work, it resembles the Lagrange-D'Alembert formulation [10, 33].

2.3. Parameter estimation

So far, the input parameters b_0, b_1 as well as the friction coefficients v_{11}, v_{12}, v_{21}, v_{22} v_{31} and v_{32} have been considered as unknown. These parameters can be estimated with 2-dimensional input-1-dimensional output pairs $((u, \dot{u}), (\phi))$ taken from real-time tests and fed into the simplex algorithm [23], which is already implemented in the Parameter Estimation Toolbox™ in MATLAB®. The algorithm minimizes a sum-of-squared error cost function $c_f = \sum(\phi - \hat{\phi})^2$ to match the real-time output data ϕ with our estimated output $\hat{\phi}$ computed with our mathematical model (2) and (6).

The signals used in the estimation process are shown in Fig. 3; they show 8 trajectories from real-time data and estimation. Rich signals are expected for estimation (not necessarily rehabilitation routines); therefore, the pair $(u, \dot{u})$ has been chosen as to produce sinusoidals of varying frequency and amplitude on ϕ. As a result, the estimated parameters in Table 2 were obtained, which completes our modelling task. As customary, a different set of data was used for model validation; these trajectories are shown in Fig. 4 and illustrate the fact that the estimated parameters are acceptable.

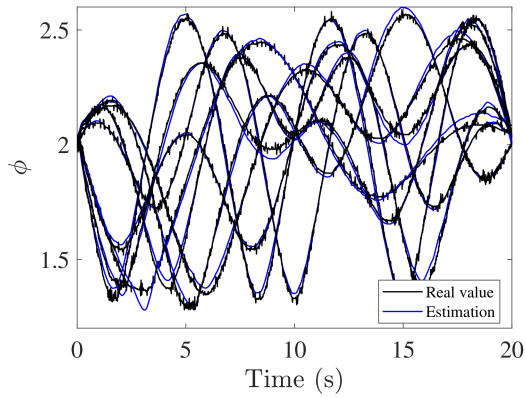


Figure 3: Trajectories estimation

Table 2: Estimated parameters using the simplex algorithm of the Parameter Estimation Toolbox in Matlab

Param.	Value	Param.	Value
v_{11}	56.3869	v_{12}	−2.8211
v_{21}	−9.3843	v_{22}	−0.6755
v_{31}	−0.0841	v_{32}	0.0440
b_0	0.0627	b_1	0.00745

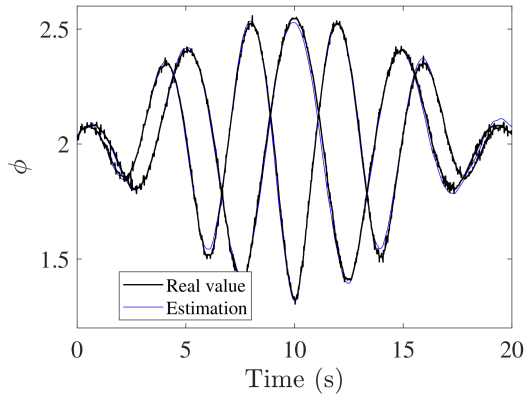


Figure 4: Validation trajectories

3. Model-based control: a computed torque proposal

The traditional form of the computed torque control is

$$\tau = M(q)\left(\ddot{q}_d - v(t)\right) + C(q,\dot{q}) + F(\dot{q}), \tag{7}$$

where $\ddot{q}_d$ corresponds to the second-time derivative of the desired trajectory q_d, and $M(q)$, $C(q,\dot{q})$ and $F(\dot{q})$ are the same as (2). Hence, as $\ddot{d}$ is the only actuated dynamic in (2) via $\dot{u}$. we must design the corresponding entry in (7), i.e.,

$$\dot{u} = b_0^{-1} M_{11}\left(\ddot{d}_d - v(t)\right) + C_1(q,\dot{q}) + F_1(\dot{q}), \tag{8}$$

with $M_{11} = m_1 + m_2 + m_3$, $C_1(q,\dot{q}) = (m_2+m_3)z_3\dot{\Theta}_1^2 + m_3 z_4\dot{\Theta}_2^2$ and $F_1(\dot{q}) = v_{11}\dot{d}^2 + v_{12}\dot{d}$.

Remark 2 *Within this section we are not using approximations of the model for design, these should be considered with the measured and estimated parameters values given in Table 1 and Table 2, respectively.*

After applying the control law (8) to the actuated dynamics in (2) a linear system for the tracking error is obtained

$$\begin{bmatrix}\dot{\epsilon}(t)\\ \dot{e}(t)\\ \ddot{e}(t)\end{bmatrix} = \begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\epsilon(t)\\ e(t)\\ \dot{e}(t)\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 1\end{bmatrix} v(t), \tag{9}$$

where the tracking error is $e(t) = d_d(t) - d(t)$ and $v(t) = k_p e(t) + k_i\epsilon(t) + k_v\dot{e}(t)$ is a PID-like control signal that can be designed via any of the methodologies for linear control with $\epsilon(t) = \int_0^t (d_d(t) - d(t))\,\mathrm{d}t$ and $\dot{e}(t) = \dot{d}_d(t) - \dot{d}(t)$.

Notice that the first time-derivative of the vector of generalized coordinates $\dot{q}$ is required for implementing the control law (8); yet, only ϕ can be measured directly, which combined with (4) only gives the position vector q to be computed. Nevertheless, taking into account the actuator dynamics (1) and the two algebraic constraints involving $\dot{q}$ in (5), it is possible to obtain

$$\dot{d} = b_1 u, \qquad \dot{\Theta}_1 = \frac{-z_4 b_1 u}{z_1 z_4 - z_2 z_3}, \qquad \dot{\Theta}_2 = \frac{z_3 b_1 u}{z_1 z_4 - z_2 z_3}, \tag{10}$$

from which it is clear that $\dot{q}$ can be computed if the control signal u is known. Importantly, this procedure avoids using observers which might compromise the control task.

To illustrate the proposed model-based control scheme with a challenging trajectory, consider the signal specified by

$$\begin{aligned} d_d(t) &= 0.05 + 0.025\sin(0.45\pi t) - 0.025\sin(0.55\pi t), \\ \dot{d}_d(t) &= 0.01125\pi\cos(0.45\pi t) - 0.01375\pi\cos(0.55\pi t), \\ \ddot{d}_d(t) &= 0.0075625\pi^2\sin(0.55\pi t) - 0.0050625\pi^2\sin(0.45\pi t), \end{aligned} \tag{11}$$

which does not correspond to a rehabilitation path. Poles of the linear error system (9) can be assigned to $s_1 = -5$, $s_2 = -6$ and $s_3 = -7$, since it was verified by simulation that these poles avoid input saturation and oscillation; using pole placement gains $k_p = -210$, $k_i = -107$, and $k_v = -18$, have been found for $v(t)$. Simulations are run from the initial conditions $q(0) = \begin{bmatrix} 0.0906 & 0.0369 & -1.9944 \end{bmatrix}^T$ and $\dot{q}(0) = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$, which are consistent with the algebraic constraints in (3) and (5), respectively. Results in Fig. 5 are obtained, which clearly indicate the control technique is able to track the desired trajectory despite its complexity.

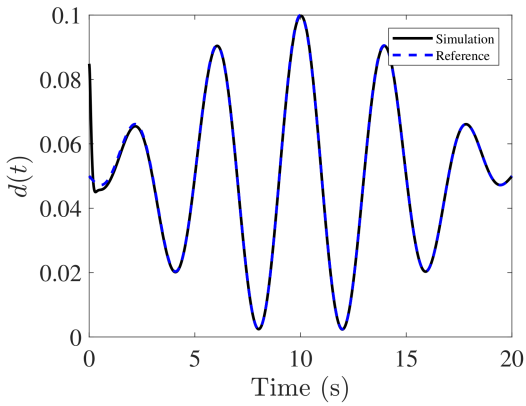


Figure 5: Time evolution for the piston and its trajectory

4. Model-free control: an ADRC proposal

If first principles or identification routines cannot be accurately performed in the motoBOTTE, model-free approaches can still be developed, as proven in this section which is concerned with an adaptation of a recently developed technique. This, of course, comes at a price: complexity of using identification and first principles in the model-based approach of the previous sections translates into the difficulty of tuning controller parameters in model-free approaches.

PID controllers are the most popular model-free technique in industrial environments, including rehabilitation engineering [15], but other approaches such as fuzzy [35] or model predictive control [1] can be found in such applications. Active disturbance rejection control (ADRC), a novel technique appeared in [17], widely acknowledged as a plausible successor of the PID as it overcomes the limitations of the latter while achieving remarkable robustness to unmodelled dynamics and disturbances, has gained a remarkable popularity in recent years. We adapt this approach to single-input parallel robots such as the motoBOTTE.

ADRC consists on the parts shown in the block diagram of Fig. 6: a transient profile generator which helps avoiding set-point jumps (therefore, not really needed in the context of rehabilitation trajectories which change continuously), an extended state observer which estimates disturbances and nonlinearities in the spirit of finite-time approaches such as sliding modes [44], and a nonlinear weighted sum which employs both the transient profile and the estimations of the observer to cancel out undesired effects as to guarantee the tracking error going to zero.

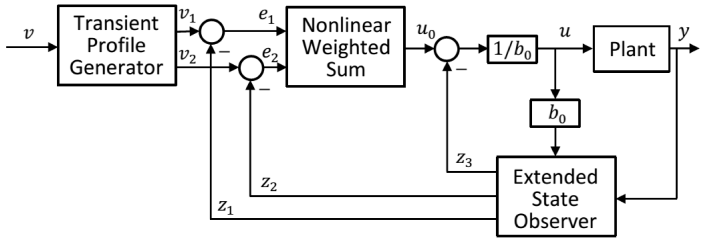


Figure 6: ADRC topology

Since the model is not used in this section, some design parameters are named after [17] in the sequel. A desired transient profile is obtained by solving the differential equations

$$\dot{v}_1 = v_2\,,$$
$$\dot{v}_2 = fhan(v_1 - v, v_2, r),$$

where v_1 is the desired trajectory, v_2 is its derivative and the function $fhan = -r\text{sign}\left(v_1 - v + \dfrac{v_2|v_2|}{2r}\right)$. The parameter r is selected to speed up or slow down the transient profile guided by the physical limitations of the plant.

The nonlinear weighted sum effectively provides feedback by nonlinearly combining PID-like laws on an interval, let say $fal(e, \alpha, \delta) = \dfrac{e}{\delta^{1-\alpha}}$ if $|e| \leqslant \delta$ with discontinuous-like ones such as $fal(e, \alpha, \delta) = |e|^{\alpha}\text{sign}(e)$, $|e| \geqslant \delta$, where $\alpha, \delta > 0$ are design parameters and $e = z_1 - y$ is an error signal.

For SISO plants total disturbance estimation and rejection is achieved by implementing $\dot{x}_1 = x_2, \dot{x}_2 = f(x_1, x_2, w(t), t) + bu$, with $y = x_1$ as the measurable output to be controlled, u being the input, and $f(\cdot)$ being a multivariable function of both the states and external disturbances. Treating $F(t) = f(x_1, x_2, w(t), t)$ as an additional state variable, $x_3 = F(t)$ and letting $\dot{F}(t) = G(t)$ with $G(t)$ unknown, the original plant is now described as $\dot{x}_1 = x_2, \dot{x}_2 = x_3 + bu, \dot{x}_3 = G(t)$, $y = x_1$, which is always observable, thus allowing the extended state observer (ESO) with equations $\dot{z}_1 = z_2 - \beta_{01}e, \dot{z}_2 = z_3 + bu - \beta_{02}fe, \dot{z}_3 = -\beta_{03}fe_1$, to be constructed, where $fe = fal(e, 0.5, \delta)$ and $fe_1 = fal(e, 0.25, \delta)$. The observer gains β_{01}, β_{02}, and β_{03} are usually chosen as $\beta_{01} = 1$, $\beta_{02} = \dfrac{1}{3h}$, $\beta_{03} = \dfrac{2}{5^2 h^{1.2}}$.

Combining the transient profile generation, the nonlinear feedback combination, and the total disturbance rejection, the ADRC control law is $u = -\dfrac{fhan(e_1, ce_2, r) + z_3}{b_0}$, where $e_1 = v_1 - z_1$ and $e_2 = v_2 - z_2$, leaving only three parameters to tune: r as the amplification coefficient that corresponds to the limit of acceleration, c as a damping coefficient to be adjusted in the neighborhood of unity, and b_0 as a rough approximation of the coefficient b in the plant within a ±50% range.

5. Real-time implementation

5.1. Rehabilitation routines

A specific and repetitive task used for rehabilitation is gait training, it can increase the strength of patient's foot and ankle [38]. Human gait is a complex movement that requires coordination of the neuro-musculo-skeletal system and it is splitted into stance and swing phases for its analysis [9, 51]. Among these phases, there are some important events where the ankle (ϕ) is involved such as: the initial contact with the floor (IC), toe landing (TL), heel off (HO), toe off (TO) and maximum dorsiflexion (MD) [5], these events are illustrated in Fig. 7, the ankle and time values for these events considering a step speed of 4 seconds are given in Table 3, see [5] for more details. This will be denoted as

the desired trajectory (ϕ_d) and can be approximated by an 11th order polynomial $\phi_d = 0.0049t^{11} - 0.1227t^{10} + 1.3103t^9 - 7.8109t^8 + 28.5069t^7 - 65.7025t^6 + 95.2514t^5 - 83.9365t^4 + 41.7027t^3 - 9.8969t^2 + 0.7538t + 1.6123$ when the inversion of a Vandermonde matrix for the indicated points is computed [7], the ankle trajectory during gait is shown in Fig. 7 with a black line.

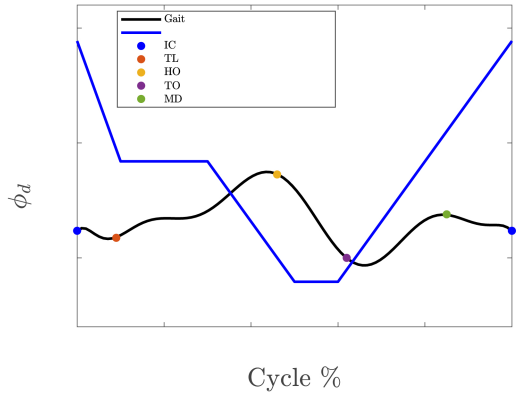


Figure 7: Ankle trajectory during gait and isokinetic exercise

Table 3: Events in the gait cycle

Event	IC	TL	HO	TO	MD
ϕ value	1.6179	1.5874	1.863	1.45	1.6895
Cycle %	0	9	49	62	89

A second trajectory is an isokinetic exercise found in [53], the reference signal has been adequated to be used with the motoBOTTE while achieving an ankle speed of 40 degrees per second [21, 52], this trajectory is shown in Fig. 7 with a blue line and it is defined as:

$$\phi_d = \begin{cases} \dfrac{\pi\,(7-2t)}{9} & t < 0.75 \\ 1.9199 & 0.75 \leqslant t < 2.25 \\ \dfrac{\pi\,(31-4t)}{36} & 2.25 \leqslant t < 3.75 \\ 1.3963 & 3.75 \leqslant t < 4.5 \\ \dfrac{\pi\,(2t-1)}{18} & 4.5 \leqslant t \leqslant 7.5. \end{cases} \tag{12}$$

5.2. Discrete-time adaptation of the control schemes

In order to implement the rehabilitation routines just described, the control schemes developed in section 3 and 4 should be translated into control signals for the motoBOTTE real-time setup. This goal requires translating continuous signals into discrete-time ones as the control laws are programmed in C language into an embedded myRIO 1900 digital device from National Instruments [19] (see Fig. 8).

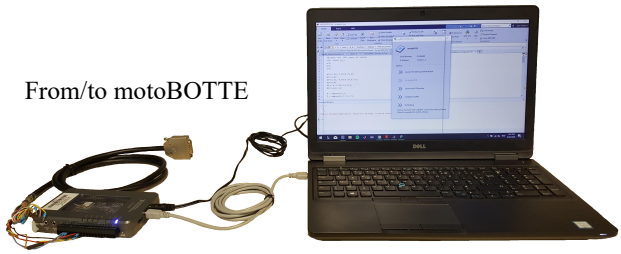


Figure 8: DSP interface

Recall that only one entry of the control law (7) is available and it corresponds to $\dot{u}$; this signal in turn depends on designing $v(t)$ in (8). By Euler's approximation [13], we have:

$$u(k+1) = u(k) + T_s \dot{u}(k), \tag{13}$$

with $u(0) = 0$. In this way, $u_(k)$ can be sent as a discrete signal into the servo controller, where $T_s = 0.01$ seconds is the sampling time for the embedded device. The servo controller is an ESCON 50/5 DC [29]; it controls the rotation speed of the DC motor inside the linear piston and is located below the foot platform of the motoBOTTE (see Fig. 1). Due to the voltage constraints of the DC motor, the control signals are limited to operate in a range of $u(k) \in [-10, 10]$.

Let us first consider the model-based computed-torque implementation. For the discrete signal $v(k)$ in section 3, consisting on three parts, Euler's approximation is used for the integral of the error

$$\epsilon(k+1) = \epsilon(k) + T_s \left(d_d(k) - d(k)\right), \tag{14}$$

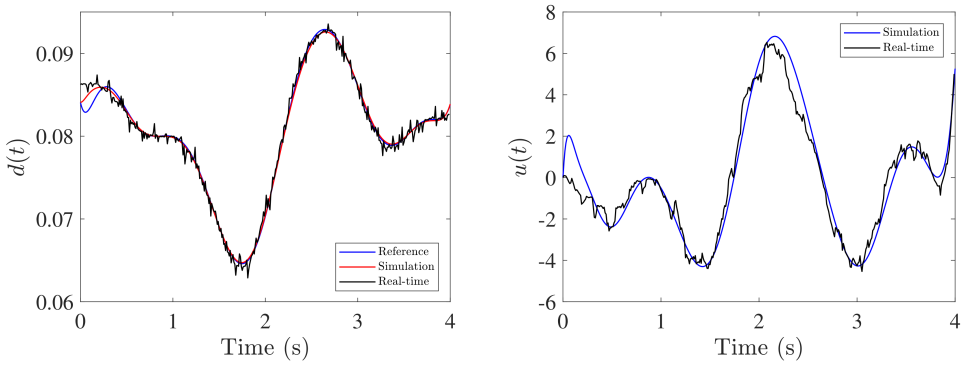
with $\epsilon(0) = 0$, while $e(k)$ and $\dot{e}(k)$ are directly available. The linear error system (9) is discretized using Tustin approximation [48], which yields

$$\begin{bmatrix} \epsilon(k+1) \\ e(k+1) \\ \dot{e}(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0.01 & 5\times 10^{-5} \\ 0 & 1 & 0.01 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \epsilon(k) \\ e(k) \\ \dot{e}(k) \end{bmatrix} + \begin{bmatrix} 2.5\times 10^{-7} \\ 5\times 10^{-5} \\ 0.01 \end{bmatrix} v(k),$$

where the signal $v(k) = k_p e(k) + k_i \epsilon(k) + k_v \dot{e}(k)$ is designed using pole placement.

The specification of the poles is usually given in continuous time; it is well-known their location is related to a variety of performance measures of the controller. The poles in continuous time have been chosen as $s_1 = -5$, $s_2 = -6$, and $s_3 = -7$, which is ensured with gains $k_p = -192.0135$, $k_i = -98.8243$, and $k_v = -16.9623$; it was verified by simulation that these poles avoid input saturation while achieving the trajectory tracking goals without oscillation. For discrete implementation, poles were translated into discrete frequency domain as $z_i = e^{s_i T_s}$ and $i \in \{1, 2, 3\}$, yielding $z_1 = 0.9512$, $z_2 = 0.9418$ and $z_3 = 0.9324$.

A first experiment intends to follow the rehabilitation routine for gait given in Fig. 7 but transformed from ankle angle $\phi(t)$ to the piston actuator coordinates $d(t)$ via (4) and (5); this trajectory is defined by real-time results obtained when tracking this reference using the discrete version of (8) with the gains above are shown in Fig. 9a, the control signal is shown in Fig. 9b; the initial conditions in simulation were $q(0) = \begin{bmatrix} 0.0841 & 0.0543 & -2.0822 \end{bmatrix}$ and $\dot{q}(0) = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$; these initial conditions must satisfy the algebraic constraints in (3) and (5) to be considered as consistent.



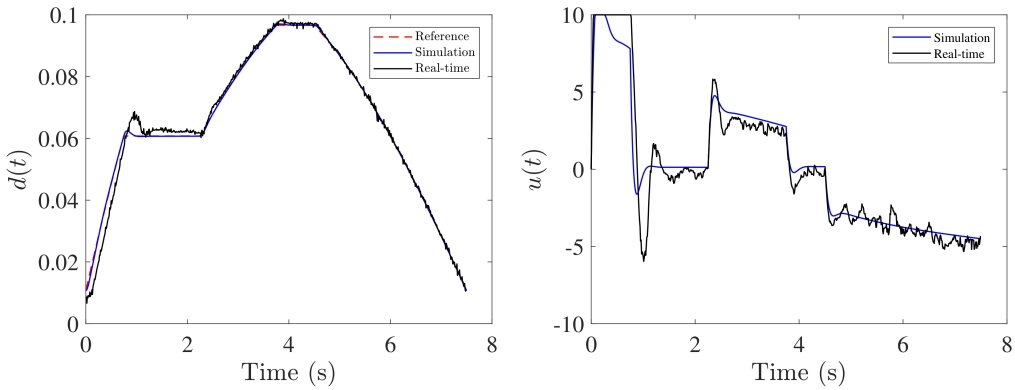
(a) Time evolution of the piston position

(b) Time evolution of the control signal

Figure 9: Gait trajectory under computed-torque control

A second experiment tracking the trajectory for isokinetic exercise (12) the trajectory is shown in Fig. 10a. The control signals applied is Fig. 10b The gains for the controller are the same as in the previous case. The consistent initial conditions for simulation were $q(0) = \begin{bmatrix} 0.0107 & 0.1574 & -2.8105 \end{bmatrix}^T$ and $\dot{q}(0) = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$.

Let us now consider the model-free ADRC implementation. The gains for the controller and the extended state observer with the input parameter were obtained using the Parameter Estimation Toolbox with the simplex algorithm [23], then,

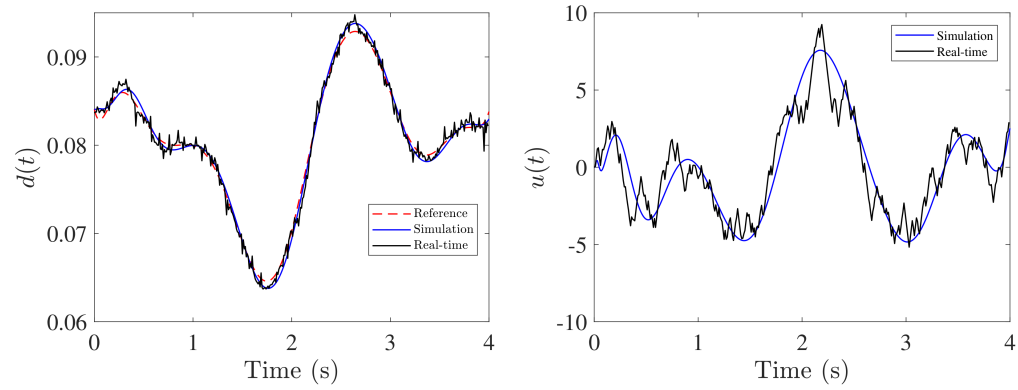


(a) Time evolution of the piston position (b) Time evolution of the control signal

Figure 10: Isokinetic exercise under computed-torque control

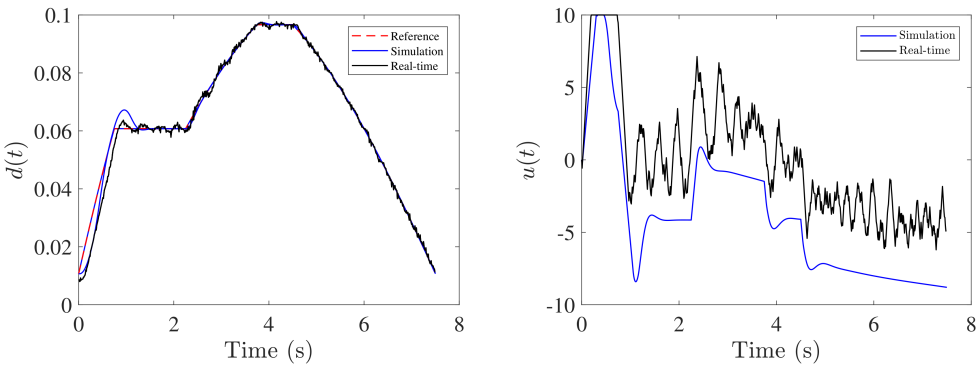
the input parameter b_2 was manually adjusted by minimizing a sum of absolute error cost function in real-time $c_{f2} = \sum |d(k) - d_d(k)|$ via bisection. The values obtained were $\beta_{01} = 0.97777$, $\beta_{02} = 2.3586$, $\beta_{03} = 0.0044$ $r_0 = 0.38843$, $c_0 = 7.5321$, $h_0 = 0.0058355$, $b_2 = 0.1088$.

The time evolution of the system when applying the active disturbance rejection control scheme with the ankle trajectory for gait is shown in Fig. 11a and its respective control signal is presented in Fig. 11b. The time evolution of the system when applying the active disturbance rejection control scheme for the isokinetic exercise is shown in Fig. 12a and its respective control signal is presented in Fig. 12b.



(a) Time evolution of the piston position (b) Time evolution of the control signal

Figure 11: Gait trajectory under ADRC



(a) Time evolution of the piston position (b) Time evolution of the control signal

Figure 12: Isokinetic exercise under ADRC

6. Discussion

From Figs. 9a and 11a it is clear that trajectory tracking for the gait rehabilitation routine is almost identical in both schemes, though the model-free ADRC proposal is slightly better, possibly due to the sliding-like characteristics of both the extended observer and the control law. A more remarkable difference emerges when control signals in Figs. 9b and 11b are compared: that of the ADRC in real-time implementation is clearly more noisy than the computed-torque control law, which is a very well-known disadvantage (or price to pay, if otherwise considered) of variable structure control [44]. In this case, chattering of the control signal does not pose any threat to the motoBOTTE piston as the actuator stands well the training cycle; yet, for a long-term use model-based computed-torque control or a smoother version of the ADRC might be advisable.

Figs. 10a and 12a prevent us from jumping to hasty conclusions about advantages and disadvantages of computed-torque and ADRC approaches: it is clear that the first one tracks the isokinetic routine better than the ADRC controller. A possible explanation lies on the less differentiability of the reference signal (it presents sudden jumps): it causes overshooting in the computed-torque trajectory tracking (see around $t = 1$ in Fig. 10a) and chattering (high-frequency signals) in Fig. 12a. This chattering is more remarkable in the control signal of the ADRC controller of Fig. 12b, which despite its wild variations realizes the tracking task; nevertheless, this is damaging for the DC motor of the piston and is not advised for practical use. On the other hand, computed-torque signal in Fig. 10b, although being more noisy than the one used for the gait routine, presents less variations than the ADRC control signal in Fig. 12b.

Based on the results just described, the control technique should be chosen according to the ankle rehabilitation routine that is going to be implemented: it

is not advisable to decide it beforehand. The model-free and model-based techniques employed in this work, namely ADRC and computed-torque adaptations, respectively, have well-known properties that have been confirmed in this implementation: model-free requires tuning against fixed design, computed-torque control signals are smoother than ADRC ones, etc. However, these characteristics may differ from the general criterium when they are applied to a particular plant (a parallel robot such as the motoBOTTE) subject to particular tasks (such as the gait and isokinetic routines).

7. Conclusions

Rehabilitation routines, namely gait cycle and isokinetic exercises, have been implemented for ankle rehabilitation on a novel parallel robotic device known as the motoBOTTE. For the sake of comparison, two control approaches, usually used for open kinematic chains, were modified in order to apply them to the motoBOTTE, which belongs the class of closed kinematic chains: model-based computed torque and model-free active disturbance rejection. It has been found that both approaches are able to perform the rehabilitation routines with slight differences in accuracy, noise, control energy, and actuator wear and tear. Remarkably, each of these techniques has been found advisable depending on the rehabilitation routine: model-based computed torque for isokinetic exercises and model-free active disturbance rejection for gait cycles.

Importantly, the mathematical model of the motoBOTTE required for the model-based computed-torque development, has been obtained in the form of differential algebraic equations; unknown parameters were estimated via identification techniques. The real-time implementations requiring a discrete-time form of the control laws as well as the hardware setup have been described in detail.

Future work may involve designing a controller that detects the patient's motion intention and generates the desired trajectory using surface electromyography signals (EMG) [45].

References

[1] N. Alibeji, N. Kirsch, S. Farrokhi, and N. Sharma: Further results on predictor-based control of neuromuscular electrical stimulation, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **23**(6), (2015), 1095–1105.

[2] J. Alvarez, J.C. Arceo, C. Armenta, J. Lauber, and M. Bernal: An extension of computed-torque control for parallel robots in ankle reeducation, *IFAC-PapersOnLine*, **52**(11), (2019), 1–6.

[3] J.C. Arceo, J. Lauber, L. Robinault, S. Paganelli, M. Jochumsen, I.K. Niazi, E. Simoneau, and S. Cremoux: Modeling and control of rehabilitation robotic device: motobotte, In *International Conference on NeuroRehabilitation*, pages 546–550. Springer, 2018.

[4] J.C. Arceo, M. Sanchez, V. Estrada-Manzo, and M. Bernal: Convex stability analysis of nonlinear singular systems via linear matrix inequalities, *IEEE Transactions on Automatic Control*, 2018.

[5] V. Arnez-Paniagua, H. Rifaï, Y. Amirat, M. Ghedira, J. M. Gracies, and S. Mohammed: Adaptive control of an actuated ankle foot orthosis for paretic patients, *Control Engineering Practice*, **90** (2019), 207–220.

[6] E.J. Benjamin, S.S. Virani, C.W. Callaway, A.M. Chamberlain, A.R. Chang, S. Cheng, S.E. Chiuve, M. Cushman, F.N. Delling, R. Deo, et al.: Heart disease and stroke statistics-2018 update: a report from the American Heart Association, *Circulation*, **137**(12), (2018), e67.

[7] Å. Björck and V. Pereyra: Solution of vandermonde systems of equations, *Mathematics of Computation*, **24**(112), (1970), 893–903.

[8] D. Brown, B. Boden-Albala, K. Langa, L. Lisabeth, M. Fair, M. Smith, R.L. Sacco, and L. Morgenstern: Projected costs of ischemic stroke in the united states, *Neurology*, **67**(8), (2006) 1390–1395.

[9] G.C. Burdea, D. Cioi, A. Kale, W.E. Janes, S.A. Ross, and J.R. Engsberg: Robotics and gaming to improve ankle strength, motor control, and function in children with cerebral palsy—a case study series, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **21**(2), (2012), 165–173.

[10] H. Cheng, Y.K. Yiu, and Z. Li: Dynamics and control of redundantly actuated parallel manipulators, *IEEE/ASME Transactions on mechatronics*, **8**(4), (2003), 483–491.

[11] D.M. Dawson, C.T. Abdallah, and F.L. Lewis: *Robot manipulator control: theory and practice*, CRC Press, 2003.

[12] I. Díaz, J. J. Gil, and E. Sánchez: Lower-limb robotic rehabilitation: literature review and challenges, *Journal of Robotics*, 2011, 2011.

[13] A. Dontchev and W. Hager: The euler approximation in state constrained optimal control, *Mathematics of Computation*, **70**(233), (2001), 173–203.

[14] V.L. Feigin, M.H. Forouzanfar, R. Krishnamurthi, G.A. Mensah, M. Connor, D.A. Bennett, A.E. Moran, R.L. Sacco, L. Anderson, T. Truelsen, et al.: Global and regional burden of stroke during 1990–2010: findings from the global burden of disease study 2010, *The Lancet*, **383**(9913), (2014), 245–255.

[15] M. Ferrarin, F. Palazzo, R. Riener, and J. Quintern: Model-based control of fes-induced single joint movements, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **9**(3), (2001), 245–257.

[16] P. Ghosh: *Numerical, Symbolic and Statistical Computing for Chemical Engineers using MATLAB*, PHI Learning Pvt. Ltd., 2018.

[17] J. Han: From pid to active disturbance rejection control, *IEEE transactions on Industrial Electronics*, **56**(3), (2009), 900–906.

[18] H. Herr: Exoskeletons and orthoses: classification, design challenges and future directions, *Journal of neuroengineering and rehabilitation*, **6**(1), (2009), 21.

[19] N. Instruments: *NI myRIO-1900 User Guide and Specifications*, National Instruments, 11500 North Mopac Expressway, Austin, Texas, 78759–3504, 376047c-01 edition, May 2016.

[20] S. Jezernik, G. Colombo, T. Keller, H. Frueh, and M. Morari: Robotic orthosis lokomat: A rehabilitation and research tool, *Neuromodulation: Technology at the neural interface*, **6**(2), (2003), 108–115.

[21] M. Jochumsen, S. Cremoux, L. Robinault, J. Lauber, J.C. Arceo, M. Navid, R. Nedergaard, U. Rashid, H. Haavik, and I. Niazi: Investigation of optimal afferent feedback modality for inducing neural plasticity with a self-paced brain-computer interface, *Sensors*, **18**(11), (2018), 3761.

[22] M.A. Khosravi and H.D. Taghirad: Robust pid control of fully-constrained cable driven parallel robots, *Mechatronics*, **24**(2), (2014), 87–97.

[23] V. Klee and G. J. Minty: How good is the simplex algorithm, Technical report, Washington Univ Seattle Dept. of Mathematics, 1970.

[24] P. Langhorne, J. Bernhardt, and G. Kwakkel: Stroke rehabilitation, *The Lancet*, **377**(9778), (2011), 1693–1702.

[25] F.L. Lewis: A survey of linear singular systems, *Circuits, Systems and Signal Processing*, **5**(1), (1986), 3–36.

[26] O. Linda and M. Manic: Uncertainty-robust design of interval type-2 fuzzy logic controller for delta parallel robot, *IEEE Transactions on Industrial Informatics*, **7**(4), (2011), 661–670.

[27] H. Markus: Stroke: causes and clinical features, *Medicine*, **36**(11), (2008), 586–591.

[28] J. Merlet: *Parallel robots*, volume 128, Springer Science & Business Media, 2006.

[29] M. Motor: *ESCON 50/5 DC Servo Controller Hardware Reference*, Maxon Motor, Bränigstrasse 220 P.O.Box 263 CH-6072 Sachseln, rel7125 edition, November 2018.

[30] N.S. Nedialkov, J.D. Pryce, and G. Tan: Algorithm 948: Daesa—a matlab tool for structural analysis of differential-algebraic equations: Software, *ACM Transactions on Mathematical Software (TOMS)*, **41**(2), (2015), 12.

[31] M. Noël, B. Cantin, S. Lambert, C.M. Gosselin, and L.J. Bouyer: An electrohydraulic actuated ankle foot orthosis to generate force fields and to test proprioceptive reflexes during human walking, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **16**(4), (2008), 390–399.

[32] C.C. Pantelides: The consistent initialization of differential-algebraic systems, *SIAM Journal on Scientific and Statistical Computing*, **9**(2), (1998), 213–231.

[33] L. Peng, Z.-G. Hou, and W. Wang: Dynamic modeling and control of a parallel upper-limb rehabilitation robot, In *2015 IEEE International Conference on Rehabilitation Robotics (ICORR)*, pages 532–537, 2015.

[34] J.C. Pérez-Ibarra and A.A. Siqueira: Comparison of kinematic and emg parameters between unassisted, fixed-and adaptive-stiffness robotic-assisted ankle movements in post-stroke subjects, In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 461–466. IEEE, 2017.

[35] N. Petroff, K.D. Reisinger, and P.A. Mason: Fuzzy-control of a hand orthosis for restoring tip pinch, lateral pinch, and cylindrical prehensions to patients with elbow flexion intact, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **9**(2), (2001), 225–231.

[36] Z. Qi, J.E. McInroy, and F. Jafari: Trajectory tracking with parallel robots using low chattering, fuzzy sliding mode controller, *Journal of Intelligent and Robotic Systems*, **48**(3), (2007) 333–356.

[37] P.J. Rabier and W.C. Rheinboldt: *Theoretical and numerical analysis of differential-algebraic equations*, Elsevier, 2002.

[38] E.J. Rouse, L.J. Hargrove, E.J. Perreault, and T.A. Kuiken: Estimation of human ankle impedance during the stance phase of walking, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **22**(4), (2014), 870–878.

[39] B.S. Rupal, S. Rafique, A. Singla, E. Singla, M. Isaksson, and G.S. Virk: Lower-limb exoskeletons: Research trends and regulatory guidelines in medical and non-medical applications, *International Journal of Advanced Robotic Systems*, **14**(6), (2017), 1729881417743554.

[40] A. Sala and C. Ariño: Polynomial fuzzy models for nonlinear control: A Taylor series approach, *IEEE Transactions on Fuzzy Systems*, **17**(6), (2009), 1284–1295.

[41] L. F. Shampine, S. Thompson, J. Kierzenka, and G. Byrne: Non-negative solutions of odes, *Applied Mathematics and Computation*, **170**(1), (2005), 556–569.

[42] W.W. Shang, S. Cong, and Y. Ge: Adaptive computed torque control for a parallel manipulator with redundant actuation, *Robotica*, **30**(3), (2012) 457–466.

[43] K.A. Shorter, G.F. Kogler, E. Loth, W.K. Durfee, and E.T. Hsiao-Wecksler: A portable powered ankle-foot orthosis for rehabilitation, *Journal of Rehabilitation Research & Development*, **48**(4), (2011).

[44] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant: *Sliding mode control and observation*, Springer, 2014.

[45] R.M. Singh, S. Chatterji, and A. Kumar: Trends and challenges in emg based control scheme of exoskeleton robots-a review, *Int. J. Sci. Eng. Res.*, **3**(9), (2012), 933–940.

[46] SKF: *CAHB-21: Linear Actuator. Installation, operation and maintenance manual*, SKF Taiwan Co., Ltd, No. 3, Lane 11, Tzu-Chiang St., Tu-Cheng Industrial District, Taipei, Taiwan, August 2010.

[47] Y. Su, B. Duan, and C. Zheng: Nonlinear pid control of a six-dof parallel manipulator, *IEE Proceedings-Control Theory and Applications*, **151**(1), (2004), 95–102.

[48] B.M. Vinagre, Y.Q. Chen, and I. Petráš: Two direct tustin discretization methods for fractional-order differentiator/integrator, *Journal of the Franklin Institute*, **340**(5), (2003), 349–362.

[49] O. Vinogradov: *Fundamentals of kinematics and dynamics of machines and mechanisms*, CRC Press, 2000.

[50] L. Wang, Z. Lu, X. Liu, K. Liu, and D. Zhang: Adaptive control of a parallel robot via backstepping technique, *International Journal of Systems, Control and Communications*, **1**(3), (2009), 312–324.

[51] D.A. Winter: *Biomechanics and motor control of human movement*, John Wiley & Sons, 2009.

[52] R. Xu, N. Jiang, N. Mrachacz-Kersting, C. Lin, G.A. Prieto, J.C. Moreno, J.L. Pons, K. Dremstrup, and D. Farina: A closed-loop brain–computer interface triggering an active ankle–foot orthosis for inducing

cortical neural plasticity, *IEEE Transactions on Biomedical Engineering*, **61**(7), (2014), 2092–2101.

[53] J. Yoon, J. Ryu, and K.-B. Lim: Reconfigurable ankle rehabilitation robot for various exercises, *Journal of Robotic Systems*, **22**(S1), (2006), S15–S33.

[54] H. Zhu, J. Doan, C. Stence, G. Lv, T. Elery, and R. Gregg: Design and validation of a torque dense, highly backdrivable powered knee-ankle orthosis, In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 504–510, IEEE, 2017.