Accepted Manuscript

International Journal of Humanoid Robotics

Article Title:	A nonlinear optimal control approach for a lower-limb robotic exoskeleton
Author(s):	G. Rigatos, M. Abbaszadeh, J. Pomares, P. Wira
DOI:	10.1142/S0219843620500188
Received:	17 October 2019
Accepted:	29 April 2020
To be cited as:	G. Rigatos *et al.*, A nonlinear optimal control approach for a lower-limb robotic exoskeleton, *International Journal of Humanoid Robotics*, doi: 10.1142/S0219843620500188
Link to final version:	https://doi.org/10.1142/S0219843620500188

World Scientific
www.worldscientific.com

A nonlinear optimal control approach for a lower-limb robotic exoskeleton

G. Rigatos[a] M. Abbaszadeh[b] J. Pomares[c] P. Wira[d]

[a]Unit of Industrial Automation
Industrial Systems Institute
26504, Rion Patras Greece
grigat@ieee.org

[b]GE Global Research
General Elecric Co.
12309, Niskayuna, NY, USA
masouda@ualberta.ca

[c] Dept. of Systems Eng.
Univ. of Alicante
03080 Alicante, Spain
jpomares@gcloud.ua.es

[d] IRIMAS
Univ. d' Haute Alsace
68093 Mulhouse, France
patrice.wira@uha.fr

Abstract: The use of robotic limb exoskeletons is growing fast either for rehabilitation purposes or in an aim to enhance human ability for lifting heavy objects or for walking for long distances without fatigue. The article proposes a nonlinear optimal control approach for a lower-limb robotic exoskeleton. The method has been successfully tested so far on the control problem of several types of robotic manipulators and the present article shows that it can also provide an optimal solution to the control problem of limb robotic exoskeletons. To implement this control scheme, the state-space model of the lower-limb robotic exoskeleton undergoes first approximate linearization around a temporary operating point, through first-order Taylor series expansion and through the computation of the associated Jacobian matrices. To select the feedback gains of the H-infinity controller an algebraic Riccati equation is solved at each time-step of the control method. The global stability properties of the control loop are proven through Lyapunov analysis. Finally, to implement state estimation-based feedback control, the H-infinity Kalman Filter is used as a robust state estimator.

Keywords: limb robotic exoskeleton, nonlinear optimal control, H-infinity control, algebraic Riccati equation, Lyapunov stability analysis, global asymptotic stability.

1 Introduction

Robotic exoskeletons find ample use in rehabilitation engineering as devices which can help impaired people to regain their gait and mobility as well as their limbs' functionality after serious injuries [1-6]. Besides, they can be used in defense and security tasks by providing to humans in military or police corps the ability to lift and transfer heavy loads or to walk for longer distances without fatigue [7-9]. The control problem of limb exoskeletons is a nontrivial one due to the nonlinear and multivariable structure of the related state-space model [10-12], So far there have been attempts to solve this control problem with the use model-based and adaptive control techniques [13-16]. Moreover, one can note efforts to solve the control problem with the use of robust control methods, that may result in abrupt variations of the control inputs [17-20]. Optimality concepts are usually not taken into account in the design of such controllers. On the one side, optimality signifies fast and accurate tracking of reference setpoints by the exoskeleton's state variables, thus improving the dexterity and precision of the performed motion. On the other side, optimality signifies minimal variations of the control inputs and minimal consumption of energy stored in batteries, thus extending the operational capacity and autonomy of exoskeletons without need to connect often to a power source [21-22].

In this article, a novel solution to the nonlinear optimal control problem of robotic lower-limb exoskele-

tons is introduced. The method relies on approximate linearization of the exoskeleton's state-space model through first-order Taylor series expansion and through the computation of the related Jacobian matrices [23-25]. The linearization takes place around a temporary operating point which is updated at each sampling period of the control algorithm and which is defined by the present value of the exoakeleton's state vector and by the most recent value of the control inputs vector. The modelling error which is due to the truncation of higher-order terms in the Taylor series expansion, is considered to be a perturbation that is asymptotically compensated by the robustness of the control scheme. For the approximately linearized model of the exoskeleton an optimal (H-infinity) feedback controller is designed [26-28].

Actually the proposed H-infinity controller stands for the solution of the optimal control problem for the robotic exoskeleton under model uncertainty and external perturbations. The H-infinity controller represents a min-max differential game in which the control inputs try to minimize a cost functional which comprises quadratic terms of the state vector's tracking error. On the other side, the model uncertainty and disturbance terms try to maximize this cost functional. To select the stabilizing gains of the H-infinity controller, an algebraic Riccati equation is solved at each sampling period of the control algorithm [29-30]. The stability properties of the control method are proven through Lyapunov analysis. First, it is shown that the control loop satisfies the H-infinity tracking performance criterion, which signifies elevated robustness against model imprecision and exogenous disturbances [1], [31-32]. Moreover, under mild conditions it is proven that the control loop is globally asymptotically stable. Finally, to perform state estimation-based control without the need to measure the entire state vector of the exoskeleton, the H-infinity Kalman Filter is used as a robust state estimator [1], [33].

The structure of the paper is as follows: In Section 2 the dynamic model of the a lower-limb exoskeleton robot is developed using Euler-Lagrange analysis. In Section 3 approximate linearization of the exoskeleton's state-space model is performed through Taylor-series expansion and the computation of Jacobian matrices around a time-varying operating point. Besides the design of an H-infinity feedback controller for the robotic exoskeleton is outlined. In Section 4 the global stability properties of the H-infinity controller for the robotic exoskeleton are proven through Lyapunov analysis. /In Section 5 The H-infinity Kalman Filter is proposed as a robust state estimator which allows for implementing state estimation-based control for the exoskeleton. In Section 6 the tracking performance of the control algorithm is further confirmed through Simulation experiments. Finally, in Section 7 concluding remarks are stated.

2 Dynamic model of the 2-DOF lower-limb exoskeleton

2.1 Dynamic model of the robotic exoskeleton

A 2-DOF lower-limb exoskeleton is considered as shown in Fig. 1. Mass m_1 is the cumulative mass of the first link and of the leg's part from hip to knee. The related cumulative moment of inertia is I_1. Mass m_2 is the cumulative mass of the second link and of the leg's part from knee to ankle. The related cumulative moment of inertia is I_2. Mass m_1 is considered to be concentrated at a distance d_1 from the first joint. The length of the first link is l_1. Mass m_2 is considered to be concentrated at a distance d_2 from the second joint. The length of the second link is l_2.

The inertial reference frame shown in Fig. 1 is considered. In this coordinates frame, the position of mass m_1 is

$$\begin{aligned} x_{m_1} &= d_1 sin(\theta_1) \\ y_{m_1} &= l_1 + l_2 - d_1 cos(\theta_1) \end{aligned} \tag{1}$$

The position of mass m_2 is given by:

$$\begin{aligned} x_{m_2} &= l_1 sin(\theta_1) + d_2 sin(\theta_1 - \theta_2) \\ y_{m_2} &= (l_1 + l_2) - l_1 cos(\theta_1) - d_2 cos(\theta_2 - \theta_1) \end{aligned} \tag{2}$$

The translational velocity of mass m_1 is

$$\begin{aligned} \dot{x}_{m_1} &= d_1 cos(\theta_1)\dot{\theta}_1 \\ \dot{y}_{m_1} &= d_1 sin(\theta_1)\dot{\theta}_1 \end{aligned} \tag{3}$$

The translational velocity of mass m_2 is

$$\begin{aligned} \dot{x}_{m_2} &= l_1 cos(\theta_1)\dot{\theta}_1 + d_2 cos(\theta_1 - \theta_2)(\dot{\theta}_1 - \dot{\theta}_2) \\ \dot{y}_{m_2} &= l_1 sin(\theta_1)\dot{\theta}_1 + d_2 sin(\theta_1 - \theta_2)(\dot{\theta}_1 - \dot{\theta}_2) \end{aligned} \tag{4}$$

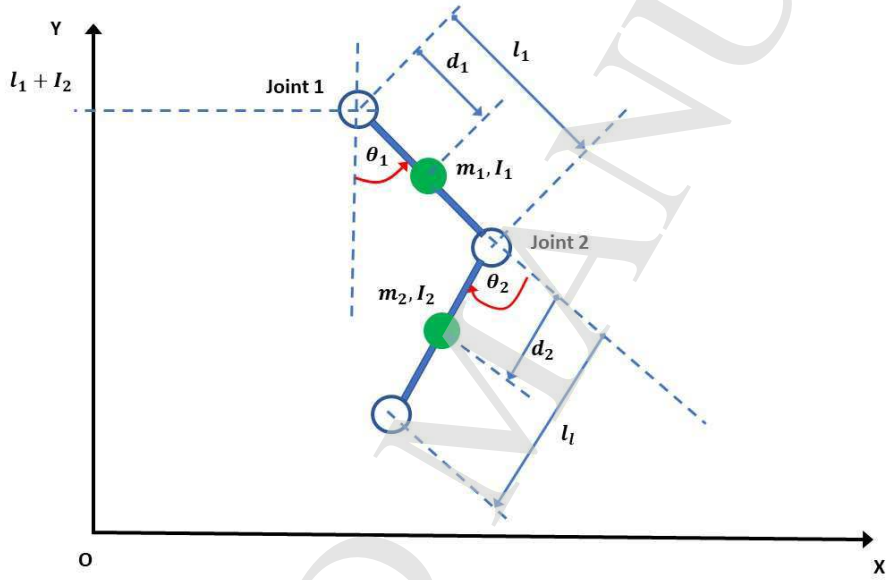


Figure 1: Diagram of a 2-DOF limb robotic exoskeeleton performing the walking motion

The kinetic energy of link 1 is

$$\begin{aligned} K_1 &= \tfrac{1}{2} m_1 (\dot{x}_{m_1}^2 + \dot{y}_{m_1}^2) + \tfrac{1}{2} I_1 \dot{\theta}_1^2 \Rightarrow \\ K_1 &= \tfrac{1}{2} m_1 [d_1^2 cos^2(\theta_1)\dot{\theta}_1^2 + d_1^2 sin^2(\theta_1)\dot{\theta}_1^2] + \tfrac{1}{2} I_1 \dot{\theta}_1^2 \Rightarrow \\ K_1 &= \tfrac{1}{2} m_1 d_1^2 \dot{\theta}_1^2 + \tfrac{1}{2} I_1 \dot{\theta}_1^2 \end{aligned} \tag{5}$$

The potential energy of link 1 is

$$P_1 = m_1 g[(l_1 + l_2) - d_1 cos(\theta_1)] \tag{6}$$

The kinetic energy of link 2 is

$$\begin{aligned} K_2 &= \tfrac{1}{2} m_1 (\dot{x}_{m_2}^2 + \dot{y}_{m_2}^2) + \tfrac{1}{2} I_1 (\dot{\theta}_1 + \dot{\theta}_2)^2 \Rightarrow \\ K_2 &= \tfrac{1}{2} m_1 [l_1^2 cos^2(\theta_1)\dot{\theta}_1^2 + d_2^2 cos^2(\theta_1 - \theta_2)(\dot{\theta}_1 - \dot{\theta}_2)^2 +] \\ &+ 2 l_1 d_2 cos(\theta_1) cos(\theta_1 - \theta_2)\dot{\theta}_1(\dot{\theta}_1 - \dot{\theta}_2) + \\ &+ \tfrac{1}{2} m_1 [l_1^2 sin^2(\theta_1)\dot{\theta}_1^2 + d_2^2 sin^2(\theta_1 - \theta_2)(\dot{\theta}_1 - \dot{\theta}_2)^2 +] \\ &+ 2 l_1 d_2 sin(\theta_1) sin(\theta_1 - \theta_2)\dot{\theta}_1(\dot{\theta}_1 - \dot{\theta}_2) + \\ &+ \tfrac{1}{2} I_1 (\dot{\theta}_1 + \dot{\theta}_2)^2 \end{aligned} \tag{7}$$

which is also written as

$$\begin{array}{c} K_2 = \frac{1}{2}m_2[l_1^2\dot{\theta}_1^2 + d_2^2(\theta_1 - \dot{\theta}_2)^2 + \\ +2l_1d_2[cos(\theta_2)cos(\theta_1 - \theta_2) + sin(\theta_2)sin(\theta_1 - \theta_2)]\dot{\theta}_1(\dot{\theta}_1 - \dot{\theta}_2)] + \\ +\frac{1}{2}I_2(\dot{\theta}_1 - \dot{\theta}_2)^2 \end{array} \tag{8}$$

After intermediate operations one obtains

$$\begin{array}{c} K_2 = \frac{1}{2}m_2[l_1^2\dot{\theta}_1^2 + d_2^2(\dot{\theta}_1 - \dot{\theta}_2^2) + 2l_1d_2cos(\theta_2)\dot{\theta}_1(\dot{\theta}_1 - \dot{\theta}_2)] + \\ +\frac{1}{2}I_2(\dot{\theta}_1 - \dot{\theta}_2)^2 \end{array} \tag{9}$$

The potential energy of link 2 is

$$P_2 = m_2g[(l_1 + l_2) - l_1cos(\theta)_1 - d_2cos(\theta_1 - \theta_2)] \tag{10}$$

The Lagrangian of the robotic exoskeleton is given by

$$L = (K_1 + K_2) - (P_1 + P_2) \tag{11}$$

Thus, the Lagrangian of the exoskeleton is given by

$$\begin{array}{c} L = [\frac{1}{2}m_1d_1^2\dot{\theta}_1^2 + \frac{1}{2}I_1\dot{\theta}_1^2] + \\ +[\frac{1}{2}m_2[l_1^2\dot{\theta}_1^2 + d_2^2(\dot{\theta}_1 - \dot{\theta}_2^2) + 2l_1d_2cos(\theta_2)\dot{\theta}_1(\dot{\theta}_1 - \dot{\theta}_2)] + \frac{1}{2}[I_2(\dot{\theta}_1 - \dot{\theta}_2)^2] - \\ -[m_1g[(l_1 + l_2) - d_1cos(\theta_1)] - \\ -[m_2g[(l_1 + l_2) - l_1cos(\theta)_1 - d_2cos(\theta_1 - \theta_2)] \end{array} \tag{12}$$

The equations of motion of the exoskeleton are obtained from the application of the Euler-Lagrange principle

$$\frac{\partial}{\partial t}\frac{\partial L}{\partial \dot{\theta}_1} - \frac{\partial L}{\partial \theta_1} = \tau_1 \tag{13}$$

$$\frac{\partial}{\partial t}\frac{\partial L}{\partial \theta_2} - \frac{\partial L}{\partial \theta_2} = \tau_2 \tag{14}$$

where τ_1 and τ_2 are the torques which are generated by the motors that make the two joints of the exoskeleton rotate. About the first link it holds that

$$\begin{array}{c} \frac{\partial L}{\partial \dot{\theta}_1} = m_1d_1^2\dot{\theta}_1 + I_1\dot{\theta}_1 + \\ +m_2l_1^2\dot{\theta}_1 + m_2d_2^2(\dot{\theta}_1 - \dot{\theta}_2) + m_2l_1d_2cos(\theta_2)(2\dot{\theta}_1 - \dot{\theta}_2) + I_2(\dot{\theta}_1 - \dot{\theta}_2) \end{array} \tag{15}$$

and also

$$\begin{array}{c} \frac{\partial}{\partial t}\frac{\partial L}{\partial \dot{\theta}_1} = m_1d_1^2\ddot{\theta}_1 + I_1\ddot{\theta}_1 + \\ +m_2l_1^2\ddot{\theta}_1 + m_2d_2^2(\ddot{\theta}_1 - \ddot{\theta}_2) - m_2l_1d_2sin(\theta_2)\dot{\theta}_2(2\dot{\theta}_1 - \dot{\theta}_2) + \\ +m_2l_1d_2cos(\theta_2)(2\ddot{\theta}_1 - \ddot{\theta}_2) + I_2(\ddot{\theta}_1 - \ddot{\theta}_2) \end{array} \tag{16}$$

while one has

$$\frac{\partial L}{\partial \theta_1} = -m_1gd_1sin(\theta_1) - m_2gl_1sin(\theta_1) - m_2gd_2sin(\theta_1 - \theta_2) \tag{17}$$

Consequently, the equation of motion of the first link becomes

$$\begin{array}{c} [m_1d_1^3 + I_1 + m_1l_1^2 + m_1d_1^2 + 2m_2l_1d_2cos(\theta)_2 + I_2]\ddot{\theta}_1 + [-m_2d_2^2 - m_2l_1d_2cos(\theta_2) - I_2]\ddot{\theta}_2 - \\ -m_2l_1d_2sin(\theta_2)\dot{\theta}_2(2\dot{\theta}_1 - \dot{\theta}_2) + (m_1gd_1 + m_2gl_2)sin(\theta_1) + m_2gd_2sin(\theta_1 - \theta_2) = \tau_1 \end{array} \tag{18}$$

About the second link it holds that

$$\frac{\partial L}{\partial \theta_2} = -m_2 d_2^2(\dot{\theta}_1 - \dot{\theta}_2) - m_2 l_1 d_2 cos(\theta)_2 \dot{\theta}_1 - I_2(\dot{\theta}_1 - \dot{\theta}_2) \tag{19}$$

and also

$$\begin{gathered}\frac{\partial}{\partial t}\frac{\partial L}{\partial \dot{\theta}_2} = -m_2 d_2^2(\ddot{\theta}_1 - \ddot{\theta}_2) + m_2 l_1 d_2 sin(\theta_2)\dot{\theta}_1\dot{\theta}_2 - \\ -m_2 l_1 d_2 cos(\theta_2)\ddot{\theta}_1 - I_1(\ddot{\theta}_1 - \ddot{\theta}_2)\end{gathered} \tag{20}$$

Moreover, one has

$$\frac{\partial L}{\partial \theta_2} = -2m_2 l_1 d_2 sin(\theta_2)\dot{\theta}_1(\dot{\theta}_1 - \dot{\theta}_2) - m_2 g d_2 sin(\theta_1 - \theta_2) \tag{21}$$

Thus, the equation of motion of the second link becomes

$$\begin{gathered}[-m_2 d_2^2 - m_2 l_1 d_2 sin(\theta_2) - I_2]\ddot{\theta}_1 + [m_2 d_2^2 + I_2]\ddot{\theta}_2 + \\ + m_2 l_1 d_2 sin(\theta_2)\dot{\theta}_1]\dot{theta}_2 + 2m_2 l_1 d_2 sin(theta_2)\dot{\theta}_1(\dot{\theta}_! - \dot{\theta}_2) + \\ m_2 g d_2 sin(\theta_1 - \theta_2) = \tau_2\end{gathered} \tag{22}$$

which is finally written as

$$\begin{gathered}[-m_2 d_2^2 - m_2 l_1 d_2 sin(\theta_2) - I_2]\ddot{\theta}_1 + [m_2 d_2^2 + I_2]\ddot{\theta}_2 + \\ 2m_2 l_1 d_2 sin(\theta_2)\dot{\theta}_1^2 + m_2 g d_2 sin(\theta_1 - \theta_2) = \tau_2\end{gathered} \tag{23}$$

In matrix form the dynamic model of the exoskeleton is written as

$$\begin{gathered}\begin{pmatrix} m_1 d_1^3 + I_1 + m_1 l_1^2 + m_1 d_1^2 + 2m_2 l_1 d_2 cos(\theta)_2 + I_2 & -m_2 d_2^2 - m_2 l_1 d_2 cos(\theta)_2 - I_2 \\ -m_2 d_2^2 - m_2 l_1 d_2 sin(\theta_2) - I_2 & m_2 d_2^2 + I_2 \end{pmatrix} \begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix} + \\ + \begin{pmatrix} -m_2 l_1 d_2 sin(\theta_2)\dot{\theta}_2(2\dot{\theta}_1 - \dot{\theta}_2) \\ 2m_2 l_1 d_2 sin(\theta_2)\dot{\theta}_1^2 \end{pmatrix} + \begin{pmatrix} (m_1 g d_1 + m_2 g l_1) sin(\theta_1) + m_2 g d_2 sin(\theta_1 - \theta_2) \\ m_2 g d_2 sin(\theta_1 - \theta_2) \end{pmatrix} = \begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix}\end{gathered} \tag{24}$$

2.2 State-space model of the robotic exoskeleton

The dynamic model of the robotic exoskeleton can be also written in the concise form

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) = \tau \tag{25}$$

where the state-vector is $\theta = [\theta_1, \theta_2]^T$, the control inputs vector is $\tau = [\tau_1, \tau_2]^T$, while the inertia matrix $M(\theta)$, the Coriolis matrix $C(\theta, \dot{\theta})$ and the gravitational forces matrix $G(\theta)$ are defined as follows:

$$M(\theta) = \begin{pmatrix} m_1 d_1^3 + I_1 + m_1 l_1^2 + m_1 d_1^2 + 2m_2 l_1 d_2 cos(\theta)_2 + I_2 & -m_2 d_2^2 - m_2 l_1 d_2 cos(\theta)_2 - I_2 \\ -m_2 d_2^2 - m_2 l_1 d_2 sin(\theta_2) - I_2 & m_2 d_2^2 + I_2 \end{pmatrix} \tag{26}$$

$$C(\theta, \dot{\theta}) \begin{pmatrix} -m_2 l_1 d_2 sin(\theta_2)\dot{\theta}_2(2\dot{\theta}_1 - \dot{\theta}_2) \\ 2m_2 l_1 d_2 sin(\theta_2)\dot{\theta}_1^2 \end{pmatrix} \quad G(\theta) = \begin{pmatrix} (m_1 g d_1 + m_2 g l_1) sin(\theta_1) + m_2 g d_2 sin(\theta_1 - \theta_2) \\ m_2 g d_2 sin(\theta_1 - \theta_2) \end{pmatrix} \tag{27}$$

Using that he inertia matrix $M(\theta)$ is

$$M = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \tag{28}$$

and that the determinant of $M(\theta)$ is $detM = M_{11}M_{22} - M_{12}^2$ the inverse of $M(\theta)$ is

$$M^{-1} = \begin{pmatrix} \frac{M_{22}}{M_{11}M_{22}-M_{12}^2} & \frac{-M_{12}}{M_{11}M_{22}-M_{12}^2} \\ \frac{-M_{21}}{M_{11}M_{22}-M_{12}^2} & \frac{M_{11}}{M_{11}M_{22}-M_{12}^2} \end{pmatrix} \tag{29}$$

Thus the dynamic model of the exoskeleton robot is written as

$$\ddot{\theta} = -M^{-1}(\theta)[C(\theta,\dot{\theta}) + G(\theta)] + M^{-1}(\theta)\tau \tag{30}$$

By performing operations between the individual matrices one arrives at the state-space description

$$\begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix} = \begin{pmatrix} \frac{-M_{22}(C_1+G_1)+M_{12}(C_2+G_2)}{M_{11}M_{22}-M_{12}^2} \\ \frac{M_{12}(C_1+G_1)-M_{11}(C_2+G_2)}{M_{11}M_{22}-M_{12}^2} \end{pmatrix} + \begin{pmatrix} \frac{M_{22}}{M_{11}M_{22}-M_{12}^2} & \frac{-M_{12}}{M_{11}M_{22}-M_{12}^2} \\ \frac{-M_{21}}{M_{11}M_{22}-M_{12}^2} & \frac{M_{11}}{M_{11}M_{22}-M_{12}^2} \end{pmatrix} \begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix} \tag{31}$$

By defining the state-vector of the exoskeleton robot as $x_1 = \theta_1$, $x_2 = \dot{\theta}_1$, $x_3 = \theta_2$, and $x_4 = \dot{\theta}_2$, the state-space model of the exoskeleton becomes

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \begin{pmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ g_{21}(x) & g_{22}(x) \\ 0 & 0 \\ g_{41}(x) & g_{42}(x) \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \tag{32}$$

where $f_1(x) = 0$, $f_2(x) = \frac{-M_{22}(C_1+G_1)+M_{12}(C_2+G_2)}{M_{11}M_{22}-M_{12}^2}$, $f_3(x) = 0$ and $f_4(x) = \frac{M_{12}(C_1+G_1)-M_{11}(C_2+G_2)}{M_{11}M_{22}-M_{12}^2}$

while $g_{11}(x) = 0$, $g_{12}(x) = 0$, $g_{21}(x) = \frac{M_{22}}{M_{11}M_{22}-M_{12}^2}$, $g_{22}(x) = \frac{-M_{12}}{M_{11}M_{22}-M_{12}^2}$, $g_{31}(x) = 0$, $g_{31}(x) = 0$, $g_{41}(x) = \frac{-M_{21}}{M_{11}M_{22}-M_{12}^2}$ and $g_{42}(x) = \frac{M_{11}}{M_{11}M_{22}-M_{12}^2}$

2.3 Differential flatness properties of the exoskeleton's model

It can be proven that the previously defined state-space model of the lower-limb robotic exoskeleton is differentially flat with flat outputs vector $Y = [x_1, x_3]^T$. This signifies that the flat outputs are the turn angles θ_1, θ_2 of the exoskeleton's joints. Actually, the differential flatness property means that (i) all state variables of the exoskeleton can be written as differential functions of the above noted flat outputs, (ii) the flat outputs are differentially independent, that is they are not connected through a relation in the form of a linear homogeneous differential equation.

From the first row of Eq. (32) of has $x_2 = \dot{x_1}$, while from the third row of the same state-space model one gets $x_4 = \dot{x}_3$. Consequent;y all state variables of the exoskeleton are expressed as differential functions of its flat outputs. From the second and fourth row of the state-space description one gets

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} g_{21}(x) & g_{22}(x) \\ g_{41}(x) & g_{42}(x) \end{pmatrix}^{-1} [\begin{pmatrix} \ddot{x}_1 \\ \ddot{x}_3 \end{pmatrix} - \begin{pmatrix} f_2(x) \\ f_4(x) \end{pmatrix}] \tag{33}$$

Eq. (33) confirms that the control inputs of the robotic exoskeleton are also differential functions of the flat outputs $Y = [x_1, x_3]^T$. This completes the proof about the differential flatness of the robotic model. The differential flatness property of the lower-limb exoskeleton can be used for defining setpoints for the related control loop.

3 Approximate linearization of the exoskeleton's dynamic model

3.1 Approximate linearization of the state-space model

The dynamic model of the robotic exoskeleton in initially written in the form

$$\dot{x} = f(x) + g(x)u \tag{34}$$

The model undergoes approximate linearization around the temporary operating point (x^*, u^*) where x^* is the present value of the exoskeleton's state vector and u^* is the most recent value of the control inputs vector. The linearization is performed through first-order Taylor series expansion and through the computation of the associated Jacobian matrices.

The linearized state-space description of the system is written as

$$\dot{x} = Ax + Bu + \tilde{d} \tag{35}$$

where $\tilde{d}$ is the cumulative vector of disturbances and modelling errors while matrices A and B are defined by the system's Jacobians

$$A = \nabla_x[f(x) + g(x)u] \mid_{(x^*,u^*)} \Rightarrow A = \nabla_x f(x) \mid_{(x^*,u^*)} + \nabla_x g(x)u \mid_{(x^*,u^*)} \tag{36}$$

$$B = \nabla_u[f(x) + g(x)u] \mid_{(x^*,u^*)} \Rightarrow B = g(x)] \mid_{(x^*,u^*)} \tag{37}$$

The computation of the Jacobian matrix $\nabla_x f(x) \mid_{(x^*,u^*)}$ is as follows:

First row of the Jacobian matrix $\nabla_x f(x) \mid_{(x^*,u^*)}$: $\frac{\partial f_1}{\partial x_1} = 0$, $\frac{\partial f_1}{\partial x_2} = 1$, $\frac{\partial f_1}{\partial x_3} = 0$ and $\frac{\partial f_1}{\partial x_4} = 0$.

Second row of the Jacobian matrix $\nabla_x f(x) \mid_{(x^*,u^*)}$: for $i = 1, 2, 3, 4$

$$\begin{aligned}\frac{\partial f_2}{\partial x_i} = & \frac{-\frac{\partial M_{22}]}{\partial x_i}(C_1+G_1) - M_{22}(\frac{\partial C_1}{\partial x_i}+\frac{\partial G_1}{\partial x_i}+\frac{\partial M_{12}]}{\partial x_i}(C_2+G_2)+M_{12}(\frac{\partial C_2}{\partial x_i}+\frac{\partial G_2}{\partial x_i})}{[M_{11}M_{22}-M_{12}^2]} - \\ & - \frac{[-M_{22}(c_1+G_1)+M_{12}(C_2+G_2)][\frac{\partial M_{11}}{\partial x_i}M_{22}+M_{11}\frac{\partial M_{22}}{\partial x_i}-2M_{12}\frac{\partial M_{12}}{\partial x_i}]}{[M_{11}M_{22}-M_{12}^2]^2}\end{aligned} \tag{38}$$

Third row of the Jacobian matrix $\nabla_x f(x) \mid_{(x^*,u^*)}$: $\frac{\partial f_3}{\partial x_1} = 0$, $\frac{\partial f_3}{\partial x_2} = 0$, $\frac{\partial f_3}{\partial x_3} = 0$ and $\frac{\partial f_3}{\partial x_4} = 1$.

Fourth row of the Jacobian matrix $\nabla_x f(x) \mid_{(x^*,u^*)}$: for $i = 1, 2, 3, 4$

$$\begin{aligned}\frac{\partial f_4}{\partial x_i} = & \frac{\frac{\partial M_{12}]}{\partial x_i}(C_1+G_1)+M_{12}(\frac{\partial C_1}{\partial x_i}+\frac{\partial G_1}{\partial x_i}+-\frac{\partial M_{11}]}{\partial x_i}(C_2+G_2)-M_{11}(\frac{\partial C_2}{\partial x_i}+\frac{\partial G_2}{\partial x_i})}{[M_{11}M_{22}-M_{12}^2]} - \\ & - \frac{[M_{12}(C_1+G_1)-M_{12}(C_2+G_2)][\frac{\partial M_{11}}{\partial x_i}M_{22}+M_{11}\frac{\partial M_{22}}{\partial x_i}-2M_{12}\frac{\partial M_{12}}{\partial x_i}]}{[M_{11}M_{22}-M_{12}^2]^2}\end{aligned} \tag{39}$$

Considering that the control inputs gain matrix consists of two column vectors, that is $g(x) = [g_1(x)\ g_2()x]$, the computation of the Jacobian matrix $\nabla_x g_1(x) \mid_{(x^*,u^*)}$ is as follows:

First row of the Jacobian matrix $\nabla_x g_1(x) \mid_{(x^*,u^*)}$: $\frac{\partial g_{11}}{\partial x_1} = 0$, $\frac{\partial g_{11}}{\partial x_2} = 0$, $\frac{\partial g_{11}}{\partial x_3} = 0$ and $\frac{\partial g_{11}}{\partial x_4} = 1$.

Second row of the Jacobian matrix $\nabla_x g_1(x) \mid_{(x^*,u^*)}$: for $i = 1, 2, 3, 4$

$$\frac{\partial g_{21}}{\partial x_i} = \frac{\frac{\partial M_{22}}{\partial x_i}}{[M_{11}M_{22}-M_{12}^2]} - \frac{M_{22}][\frac{\partial M_{11}}{\partial x_i}M_{22}+M_{11}\frac{\partial M_{22}}{\partial x_i}-2M_{12}\frac{\partial M_{12}}{\partial x_i}}{[M_{11}M_{22}-M_{12}^2]^2} \tag{40}$$

Third row of the Jacobian matrix $\nabla_x g_1(x)\mid_{(x^*,u^*)}$: $\frac{\partial g_{31}}{\partial x_1}=0$, $\frac{\partial g_{31}}{\partial x_2}=0$, $\frac{\partial g_{31}}{\partial x_3}=0$ and $\frac{\partial g_{31}}{\partial x_4}=1$.

Fourth row of the Jacobian matrix $\nabla_x g_1(x)\mid_{(x^*,u^*)}$: for $i=1,2,3,4$

$$\frac{\partial g_{41}}{\partial x_i}=\frac{-\frac{\partial M_{12}}{\partial x_i}}{[M_{11}M_{22}-M_{12}^2]}-\frac{-M_{12}][\frac{\partial M_{11}}{\partial x_i}M_{22}+M_{11}\frac{\partial M_{22}}{\partial x_i}-2M_{12}\frac{\partial M_{12}}{\partial x_i}}{[M_{11}M_{22}-M_{12}^2]^2} \tag{41}$$

The computation of the Jacobian matrix $\nabla_x g_2(x)\mid_{(x^*,u^*)}$ is as follows:

First row of the Jacobian matrix $\nabla_x g_2(x)\mid_{(x^*,u^*)}$: $\frac{\partial g_{12}}{\partial x_1}=0$, $\frac{\partial g_{12}}{\partial x_2}=0$, $\frac{\partial g_{12}}{\partial x_3}=0$ and $\frac{\partial g_{12}}{\partial x_4}=1$.

Second row of the Jacobian matrix $\nabla_x g_2(x)\mid_{(x^*,u^*)}$: for $i=1,2,3,4$

$$\frac{\partial g_{22}}{\partial x_i}=\frac{-\frac{\partial M_{12}}{\partial x_i}}{[M_{11}M_{22}-M_{12}^2]}-\frac{-M_{12}][\frac{\partial M_{11}}{\partial x_i}M_{22}+M_{11}\frac{\partial M_{22}}{\partial x_i}-2M_{12}\frac{\partial M_{12}}{\partial x_i}}{[M_{11}M_{22}-M_{12}^2]^2} \tag{42}$$

Third row of the Jacobian matrix $\nabla_x g_2(x)\mid_{(x^*,u^*)}$: $\frac{\partial g_{32}}{\partial x_1}=0$, $\frac{\partial g_{32}}{\partial x_2}=0$, $\frac{\partial g_{32}}{\partial x_3}=0$ and $\frac{\partial g_{32}}{\partial x_4}=1$.

Fourth row of the Jacobian matrix $\nabla_x g_2(x)\mid_{(x^*,u^*)}$: for $i=1,2,3,4$

$$\frac{\partial g_{42}}{\partial x_i}=\frac{\frac{\partial M_{11}}{\partial x_i}}{[M_{11}M_{22}-M_{12}^2]}-\frac{M_{11}][\frac{\partial M_{11}}{\partial x_i}M_{22}+M_{11}\frac{\partial M_{22}}{\partial x_i}-2M_{12}\frac{\partial M_{12}}{\partial x_i}}{[M_{11}M_{22}-M_{12}^2]^2} \tag{43}$$

Next, the partial derivatives of the individual elements of the inertia, Coriolis and gravitational matrices are computed. It holds that:

Inertia matrix $M(\theta)$:

$\frac{\partial M_{11}}{\partial x_1}=0$, $\frac{\partial M_{11}}{\partial x_2}=0$, $\frac{\partial M_{11}}{\partial x_3}=-2m_2l_1d_2sin(x_3)$, $\frac{\partial M_{11}}{\partial x_4}=0$.

$\frac{\partial M_{12}}{\partial x_1}=0$, $\frac{\partial M_{12}}{\partial x_2}=0$, $\frac{\partial M_{12}}{\partial x_3}=m_2l_1d_2sin(x_3)$, $\frac{\partial M_{12}}{\partial x_4}=0$.

$\frac{\partial M_{22}}{\partial x_1}=0$, $\frac{\partial M_{22}}{\partial x_2}=0$, $\frac{\partial M_{22}}{\partial x_3}=0$, $\frac{\partial M_{22}}{\partial x_4}=0$.

Coriolis matrix $C(\theta,\dot{\theta})$:

$\frac{\partial C_1}{\partial x_1}=0$, $\frac{\partial C_1}{\partial x_2}=-2m_2l_1d_2sin(x_3)x_4$, $\frac{\partial C_1}{\partial x_3}=-m_2l_1d_2cos(x_3)x_4(2x_2-x_4)$, $\frac{\partial C_1}{\partial x_4}=-m_2l_1d_2sin(x_3)(2x_2-x_4)+m_2l_1d_2sin(x_3)x_4$.

$\frac{\partial C_2}{\partial x_1}=0$, $\frac{\partial C_2}{\partial x_2}=2m_2l_1d_2sin(x_3)x_2$, $\frac{\partial C_2}{\partial x_3}=2m_2l_1d_2cos(x_3)x_2^2$, $\frac{\partial C_2}{\partial x_4}=0$

Gravitational martix $G(\theta)$:

$\frac{\partial G_1}{\partial x_1}=(m_1gd_1+m_2gl_1)cos(x_1)+m_2gd_2sin(x_1-x_3)$, $\frac{\partial G_1}{\partial x_2}=0$, $\frac{\partial G_1}{\partial x_3}=-m_2gd_2cos(x_1-x_3)$, $\frac{\partial G_1}{\partial x_4}=0$.

$\frac{\partial G_2}{\partial x_1}=m_2gd_2cos(x_1-x_3)$, $\frac{\partial G_1}{\partial x_2}=0$, $\frac{\partial G_1}{\partial x_3}=-m_2gd_2cos(x_1-x_3)$, $\frac{\partial G_1}{\partial x_4}=0$.

3.2 Stabilizing feedback control

After linearization around its current operating point, the dynamic model of the lower-limb exoskeleton robot is written as

$$\dot{x}=Ax+Bu+d_1 \tag{44}$$

Parameter d_1 stands for the linearization error in the lower-limb exoskeleton robot's dynamic model appearing previously in Eq. (35). The reference setpoints for the lower-limb exoskeleton robot's state vector are denoted by $\mathbf{x_d} = [x_1^d, \cdots, x_4^d]$. Tracking of this trajectory is achieved after applying the control input u^*. At every time instant the control input u^* is assumed to differ from the control input u appearing in Eq. (44) by an amount equal to Δu, that is $u^* = u + \Delta u$

$$\dot{x}_d = Ax_d + Bu^* + d_2 \tag{45}$$

The dynamics of the controlled system described in Eq. (44) can be also written as

$$\dot{x} = Ax + Bu + Bu^* - Bu^* + d_1 \tag{46}$$

and by denoting $d_3 = -Bu^* + d_1$ as an aggregate disturbance term one obtains

$$\dot{x} = Ax + Bu + Bu^* + d_3 \tag{47}$$

By subtracting Eq. (45) from Eq. (47) one has

$$\dot{x} - \dot{x}_d = A(x - x_d) + Bu + d_3 - d_2 \tag{48}$$

By denoting the tracking error as $e = x - x_d$ and the aggregate disturbance term as $\tilde{d} = d_3 - d_2$, the tracking error dynamics becomes

$$\dot{e} = Ae + Bu + \tilde{d} \tag{49}$$

For the approximately linearized model of the system a stabilizing feedback controller is developed. The controller has the form

$$u(t) = -Ke(t) \tag{50}$$

with $K = \frac{1}{r}B^T P$ where P is a positive definite symmetric matrix which is obtained from the solution of the Riccati equation [1]

$$A^T P + PA + Q - P(\tfrac{2}{r}BB^T - \tfrac{1}{\rho^2}LL^T)P = 0 \tag{51}$$

where Q is a positive semi-definite symmetric matrix. The diagram of the considered control loop is depicted in Fig. 2

4 Lyapunov stability analysis

Through Lyapunov stability analysis it will be shown that the proposed nonlinear control scheme assures H_∞ tracking performance for the lower-limb robotic exoskeleton, and that in case of bounded disturbance terms asymptotic convergence to the reference setpoints is achieved. The tracking error dynamics for the robotic exoskeleton is written in the form

$$\dot{e} = Ae + Bu + L\tilde{d} \tag{52}$$

where in the limb robotic exoskeleton's case $L = I \in R^4$ with I being the identity matrix. Variable $\tilde{d}$ denotes model uncertainties and external disturbances of the robotic exoskeleton's model. The following Lyapunov equation is considered

$$V = \tfrac{1}{2}e^T Pe \tag{53}$$

where $e = x - x_d$ is the tracking error. By differentiating with respect to time one obtains

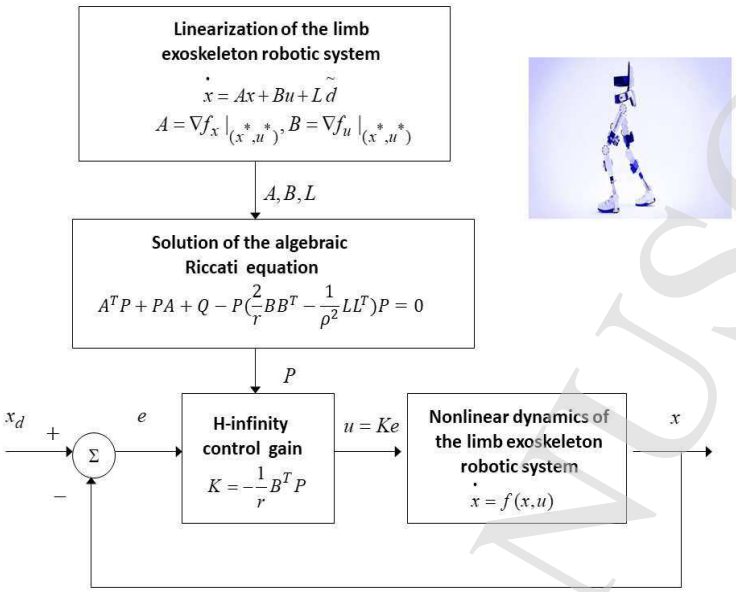


Figure 2: Diagram of the control scheme for the lower-limb robotic exoskeleton

$$\dot{V} = \tfrac{1}{2}\dot{e}^T P e + \tfrac{1}{2} e P \dot{e} \Rightarrow$$
$$\dot{V} = \tfrac{1}{2}[Ae + Bu + L\tilde{d}]^T P e + \tfrac{1}{2} e^T P [Ae + Bu + L\tilde{d}] \Rightarrow \quad (54)$$

$$\dot{V} = \tfrac{1}{2}[e^T A^T + u^T B^T + \tilde{d}^T L^T] P e +$$
$$+ \tfrac{1}{2} e^T P [Ae + Bu + L\tilde{d}] \Rightarrow \quad (55)$$

$$\dot{V} = \tfrac{1}{2} e^T A^T P e + \tfrac{1}{2} u^T B^T P e + \tfrac{1}{2}\tilde{d}^T L^T P e +$$
$$\tfrac{1}{2} e^T P A e + \tfrac{1}{2} e^T P B u + \tfrac{1}{2} e^T P L \tilde{d} \quad (56)$$

The previous equation is rewritten as

$$\dot{V} = \tfrac{1}{2} e^T (A^T P + PA) e + (\tfrac{1}{2} u^T B^T P e + \tfrac{1}{2} e^T P B u) +$$
$$+ (\tfrac{1}{2}\tilde{d}^T L^T P e + \tfrac{1}{2} e^T P L \tilde{d}) \quad (57)$$

Assumption: For given positive definite matrix Q and coefficients r and ρ there exists a positive definite matrix P, which is the solution of the following matrix equation

$$A^T P + PA = -Q + P(\tfrac{2}{r} BB^T - \tfrac{1}{\rho^2} LL^T) P \quad (58)$$

Moreover, the following feedback control law is applied to the system

$$u = -\tfrac{1}{r} B^T P e \quad (59)$$

By substituting Eq. (58) and Eq. (59) one obtains

$$\dot{V} = \tfrac{1}{2} e^T [-Q + P(\tfrac{2}{r} BB^T - \tfrac{1}{\rho^2} LL^T) P] e +$$
$$+ e^T P B (-\tfrac{1}{r} B^T P e) + e^T P L \tilde{d} \Rightarrow \quad (60)$$

$$\dot{V} = -\tfrac{1}{2} e^T Q e + \tfrac{1}{r} e^T P B B^T P e - \tfrac{1}{2\rho^2} e^T P L L^T P e$$
$$- \tfrac{1}{r} e^T P B B^T P e + e^T P L \tilde{d} \quad (61)$$

which after intermediate operations gives

$$\dot{V} = -\tfrac{1}{2}e^T Qe - \tfrac{1}{2\rho^2}e^T PLL^T Pe + e^T PL\tilde{d} \tag{62}$$

or, equivalently

$$\begin{array}{c}\dot{V} = -\tfrac{1}{2}e^T Qe - \tfrac{1}{2\rho^2}e^T PLL^T Pe + \\ +\tfrac{1}{2}e^T PL\tilde{d} + \tfrac{1}{2}\tilde{d}^T L^T Pe\end{array} \tag{63}$$

Lemma: The following inequality holds

$$\tfrac{1}{2}e^T L\tilde{d} + \tfrac{1}{2}\tilde{d}L^T Pe - \tfrac{1}{2\rho^2}e^T PLL^T Pe \leq \tfrac{1}{2}\rho^2\tilde{d}^T\tilde{d} \tag{64}$$

Proof: The binomial $(\rho\alpha - \frac{1}{\rho}b)^2$ is considered. Expanding the left part of the above inequality one gets

$$\begin{array}{c}\rho^2a^2 + \frac{1}{\rho^2}b^2 - 2ab \geq 0 \Rightarrow \tfrac{1}{2}\rho^2a^2 + \tfrac{1}{2\rho^2}b^2 - ab \geq 0 \Rightarrow \\ ab - \tfrac{1}{2\rho^2}b^2 \leq \tfrac{1}{2}\rho^2a^2 \Rightarrow \tfrac{1}{2}ab + \tfrac{1}{2}ab - \tfrac{1}{2\rho^2}b^2 \leq \tfrac{1}{2}\rho^2a^2\end{array} \tag{65}$$

The following substitutions are carried out: $a = \tilde{d}$ and $b = e^T PL$ and the previous relation becomes

$$\tfrac{1}{2}\tilde{d}^T L^T Pe + \tfrac{1}{2}e^T PL\tilde{d} - \tfrac{1}{2\rho^2}e^T PLL^T Pe \leq \tfrac{1}{2}\rho^2\tilde{d}^T\tilde{d} \tag{66}$$

Eq. (66) is substituted in Eq. (63) and the inequality is enforced, thus giving

$$\dot{V} \leq -\tfrac{1}{2}e^T Qe + \tfrac{1}{2}\rho^2\tilde{d}^T\tilde{d} \tag{67}$$

Eq. (67) shows that the H_∞ tracking performance criterion is satisfied. The integration of $\dot{V}$ from 0 to T gives

$$\begin{array}{c}\int_0^T \dot{V}(t)dt \leq -\tfrac{1}{2}\int_0^T ||e||_Q^2 dt + \tfrac{1}{2}\rho^2\int_0^T ||\tilde{d}||^2 dt \Rightarrow \\ 2V(T) + \int_0^T ||e||_Q^2 dt \leq 2V(0) + \rho^2\int_0^T ||\tilde{d}||^2 dt\end{array} \tag{68}$$

Moreover, if there exists a positive constant $M_d > 0$ such that $\int_0^\infty ||\tilde{d}||^2 dt \leq M_d$, then one gets

$$\int_0^\infty ||e||_Q^2 dt \leq 2V(0) + \rho^2 M_d \tag{69}$$

Thus, the integral $\int_0^\infty ||e||_Q^2 dt$ is bounded. Moreover, $V(T)$ is bounded and from the definition of the Lyapunov function V in Eq. (53) it becomes clear that $e(t)$ will be also bounded since $e(t) \in \Omega_e = \{e | e^T Pe \leq 2V(0) + \rho^2 M_d\}$. According to the above and with the use of Barbalat's Lemma one obtains $\lim_{t\to\infty} e(t) = 0$.

The outline of the global stability proof is that at each iteration of the control algorithm the state vector of the lower-limb robotic exoskeleton converges towards the temporary equilibrium and the temporary equilibrium in turn converges towards the reference trajectory. Thus, the control scheme exhibits global asymptotic stability properties and not local stability. Assume the i-th iteration of the control algorithm and the i-th time interval about which a positive definite symmetric matrix P is obtained from the solution of the Riccati Equation appearing in Eq. (58). By following the stages of the stability proof one arrives at Eq. (67) which shows that the H-infinity tracking performance criterion holds. By selecting the attenuation coefficient ρ to be sufficiently small and in particular to satisfy $\rho^2 < ||e||_Q^2 / ||\tilde{d}||^2$ one has that the first derivative of the Lyapunov function is upper bounded by 0. Therefore for the i-th time interval it is proven that the Lyapunov function defined in Eq (53) is a decreasing one. This signifies that between the beginning and the end of the i-th time interval there will be a drop of the value of the Lyapunov function and since matrix P is a positive definite one, the only way for this to happen is the Euclidean norm of the state vector error e to be decreasing. This means that comparing to the beginning of each time interval,

the distance of the state vector error from 0 at the end of the time interval has diminished. Consequently as the iterations of the control algorithm advance the tracking error will approach zero, and this is a global asymptotic stability condition.

5 State estimation with robust Kalman Filtering

The control loop has to be implemented with the use of information provided by a small number of sensors and by processing only a small number of state variables. To reconstruct the missing information about the state vector of the lower-limb robotic exoskeleton it is proposed to use a filtering scheme and based on it to apply state estimation-based control [1], [30]. The recursion of the H_∞ Kalman Filter, for the model of robotic exoskeleton, can be formulated in terms of a *measurement update* and a *time update* part

Measurement update:

$$\begin{aligned} D(k) &= [I - \theta W(k)P^{-}(k) + C^{T}(k)R(k)^{-1}C(k)P^{-}(k)]^{-1} \\ K(k) &= P^{-}(k)D(k)C^{T}(k)R(k)^{-1} \\ \hat{x}(k) &= \hat{x}^{-}(k) + K(k)[y(k) - C\hat{x}^{-}(k)] \end{aligned} \tag{70}$$

Time update:

$$\begin{aligned} \hat{x}^{-}(k+1) &= A(k)x(k) + B(k)u(k) \\ P^{-}(k+1) &= A(k)P^{-}(k)D(k)A^{T}(k) + Q(k) \end{aligned} \tag{71}$$

where it is assumed that parameter θ is sufficiently small to assure that the covariance matrix $P^{-}(k)^{-1} - \theta W(k) + C^{T}(k)R(k)^{-1}C(k)$ will be positive definite. When $\theta = 0$ the H_∞ Kalman Filter becomes equivalent to the standard Kalman Filter. One can measure only a part of the state vector of the lower-limb robotic exoskeleton, for instance state variables $x_1 = \theta_1$ and $x_3 = \theta_2$, and can estimate through filtering the rest of the state vector elements (angular velocities of the joints). Moreover, the proposed Kalman filtering method can be used for sensor fusion purposes.

6 Simulation tests

The tracking performance of the proposed nonlinear optimal (H-infinity) control scheme for the robotic exoskeleton's model has been further confirmed through simulation experiments. The obtained results are depicted in Fig. 3 to Fig. 8 It can be noticed, that in all cases fast and accurate tracking of the reference setpoints was achieved under moderate variations of the control inputs. The real values of the state-variables of the exoskeleton are depicted in blue, their estimated values are printed in green, while the associated setpoints are plotted in red. The tracking accuracy of the control method as well as its transient performance depends on the selection of the gains r, ρ, and of matrix Q which appears in the algebraic Riccati equation of Eq. (58). The necessary and sufficient condition for the application of the proposed nonlinear optimal control scheme is the existence of a solution of the aforementioned Riccati equation. The robustness of the control scheme depends on the selection of the attenuation coefficient ρ. Actually, the smallest value of ρ for which a solution of this Riccati equation can be obtained is the one that provides maximum robustness to the control loop.

Comparing to other nonlinear control methods for the exoskeleton's model, the article's nonlinear optimal (H-infinity) control scheme exhibits specific advantages: (i) unlike global linearization-based control approaches (for instance Lie algebra-based control or differential flatness theory-based control) the article's method does not require complicated changes of state variables (diffeomorphisms) for bringing the system into an equivalent linearized form (ii) unlike global linearization-based control approaches in the article's method the control input is applied directly on the initial nonlinear state-space model of the robotic system

and does not require inverse transformation. In this manner singularity problems are avoided (iii) unlike popular approaches for optimal control being used in industry, for instance MPC or NMPC the article's method is of proven global stability and the conference of its iterative search for an optimum is also assured. It is noted that MPC is developed for linear systems and its use in the nonlinear state-space model of the exoskeleton will result into loss of stability. Besides, the performance of NMPC depends on initialization and parameter values selection. (iv) unlike backstepping control the article's control method can be applie to a very wide class of nonlinear dynamical systems without requiring the related state-space description to be found in a specific state-space form (for instance the triangular or backstepping integral form) (v) unlike sliding-mode control the article's control scheme does not require the definition of sliding surfaces. It is noted that the definition of sliding surface can be an ad-hoc procedure if the state-space description of the system is not found in the canonical (input-output linearized) form (vi) unlike PID control the article's control method is of assured global stability, does not require heuristic tuning of parameters and functions reliably under changes of operating points. (vii) unlike multiple models-based control and linearization of the exoskeleton's dynamics around multiple operating points, the article's method requires linearization only around one operating point and needs the solution of only one algebraic Riccati equation. Consequently the article's method is computationally more efficient and its stability properties do not rely on the co-existence of solutions for multiple Riccati equations.

7 Conclusions

The wide deployment of the use of exoskeletons for rehabilitation or defense purposes makes also necessary the development of efficient control algorithms about them. To this end, the present article has proposed a novel nonlinear optimal (H-infinity) control scheme which improves the dexterity and agility of the operations performed by exoskeletons, while also raising the operational capacity and autonomy of these robotic systems. In this new control approach, the dynamic model of exoskeletons undergoes first approximate linearization with the use of first-order Taylor series expansion and through the computation of the related Jacobian matrices. The time-varying operating point is updated at each time-step of the control algorithm and is defined by the present value of the exoskeleton's state vector and by the most recent value of the control inputs vector.

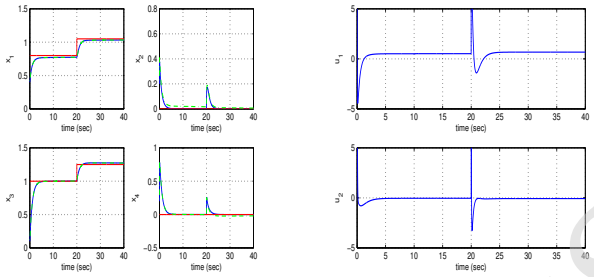
For the approximately linearized model of the exoskeleton an H-infinity (optimal) feedback controller has been designed. The controller stands for the solution of the exoskeleton's optimal control problem under model uncertainty and external perturbations. To select the controller's feedback gains an algebraic Riccati equation is solved at each sampling period of the control method. The stability properties of the control scheme are proven through Lyapunov analysis. Under mild conditions it is demonstrated that the exoskeleton's control loop is globally asymptotically stable. Finally, by using the H-infinity Kalman Filter as a state estimator it is shown that state estimation-based control can be applied to the exoskeleton without need to measure its entire state vector.

References

[1] G. Rigatos and K. Busawon, Robotic manipulators and vehicle: Control, estimation and filtering, Springer, 2018.

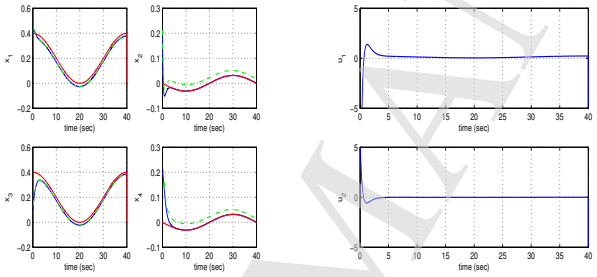
[2] A.U. Pehlivan, J.P. Lasey and M.K. O' Malley, Minimal assist-as-needed controller for upper limb robotic rehabilitation, IEEE Transactions on Robotics, vol. 32, no. 1, pp. 113-124, 2016.

[3] S. Mefoued, S. Mohammad and Y. Amirat, Toward movement restoration of knee joint using robust control of powered orthosis, IEEE Transactions on Control Systems Technology, vol. 31, no. 6, pp. 2156-2168, 2013



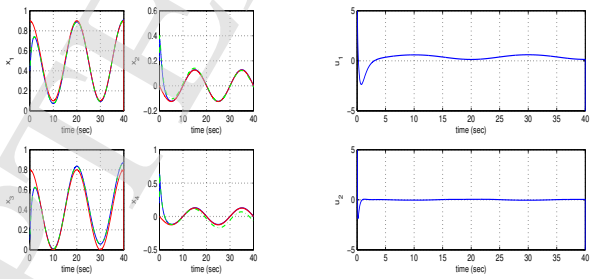
(a) **(b)**

Figure 3: Tracking of setpoint 1 by the robotic exoskeleton: (a) Convergence of the state variables of the exoskeleton $x_i,\ i = 1, \cdots, 4$ (blue line) to the associated reference values (red line) and estimated values (green line) (b) Control inputs $u_i,\ i = 1, 2$ (motor torques) applied to the exoskeleton



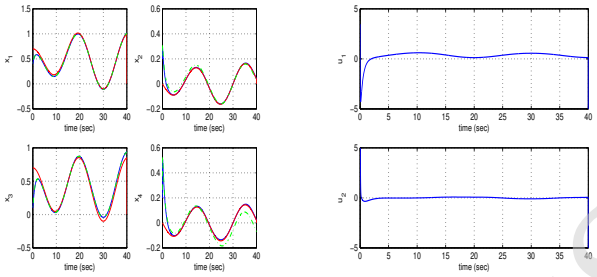
(a) **(b)**

Figure 4: Tracking of setpoint 2 by the robotic exoskeleton: (a) Convergence of the state variables of the exoskeleton $x_i,\ i = 1, \cdots, 4$ (blue line) to the associated reference values (red line) and estimated values (green line) (b) Control inputs $u_i,\ i = 1, 2$ (motor torques) applied to the exoskeleton



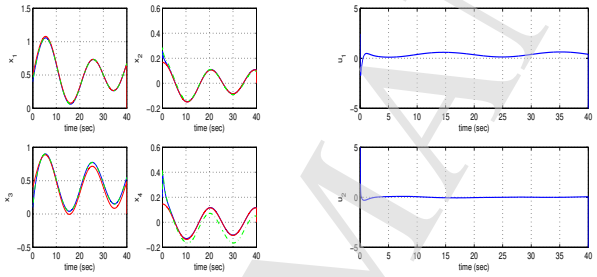
(a) **(b)**

Figure 5: Tracking of setpoint 3 by the robotic exoskeleton: (a) Convergence of the state variables of the exoskeleton $x_i,\ i = 1, \cdots, 4$ (blue line) to the associated reference values (red line) and estimated values (green line) (b) Control inputs $u_i,\ i = 1, 2$ (motor torques) applied to the exoskeleton



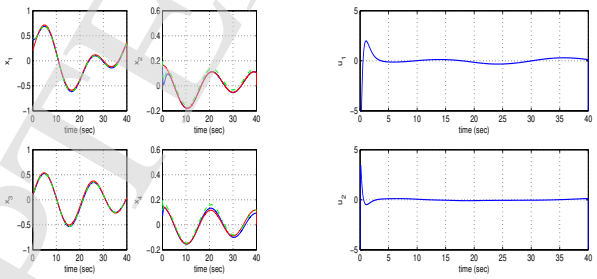
(a) **(b)**

Figure 6: Tracking of setpoint 4 by the robotic exoskeleton: (a) Convergence of the state variables of the exoskeleton $x_i,\ i = 1, \cdots, 4$ (blue line) to the associated reference values (red line) and estimated values (green line) (b) Control inputs $u_i,\ i = 1, 2$ (motor torques) applied to the exoskeleton



(a) **(b)**

Figure 7: Tracking of setpoint 5 by the robotic exoskeleton: (a) Convergence of the state variables of the exoskeleton $x_i,\ i = 1, \cdots, 4$ (blue line) to the associated reference values (red line) and estimated values (green line) (b) Control inputs $u_i,\ i = 1, 2$ (motor torques) applied to the exoskeleton



(a) **(b)**

Figure 8: Tracking of setpoint 6 by the robotic exoskeleton: (a) Convergence of the state variables of the exoskeleton $x_i,\ i = 1, \cdots, 4$ (blue line) to the associated reference values (red line) and estimated values (green line) (b) Control inputs $u_i,\ i = 1, 2$ (motor torques) applied to the exoskeleton

[4] P. Yang, X. Ma, J. Wang, G. Zhang, Y. Zhang and L. Chan, Disturbance observer-based sliding-mode control of a 5-DOF upper-limb exoskeleton robot, IEEE Access, vol. 7, pp. 6283-62839, 2019,

[5] M. Grun and O. Konigorski, Observer-based method for joint torque estimation in active orthosis, MathMod 2012, 7th Vienna Intl. Conf. on Mathematical Modelling, Vienna, Austria, Feb. 2012

[6] S. Yu, H. Lee, W. Kim and C. Han, Development of an underactuated exoskeleton for effective walking and load-carrying assist, Advanced Robotics, Taylor and Francis, vol. 30, no. 8, pp. 535-551, 2016.

[7] B. Huang, Z. Li, X. Wu, A Ajoudani, A. Bicchi and J. Liu, Coordination control of a dual-arm exoskeleton robot using human inpedance trasfer skills, IEEE Transactions on Systems, Man and Cybernetics: Systems, vol. 49, no. 5, pp. 954-963, 2019

[8] S. Chen, Z. Chen and B. Yao, Precision cascade frce control of multi-DOF hydraulic leg exoskeleton, IEEE Access vol. 6, pp. 8574-8583, 2018.

[9] G. Lv, M. Zhou and R.D. Gregg, A discussion of past and ongoing work on the design and control of highly back-drivable lower-limb exoskeleton, IEEE Control Systems Magazine, pp. 88-113, Dec. 2018

[10] Z. Chen, Z. Li and C.L.P. Chen, Disturbance observer-based fuzzy control of uncertain MIMO mechanical systems with input nonlinearities and its application to robotic exoskeleton, IEEE Transactions on Cybernetics, vol. 47, no. 4, pp. 984-994, 2012.

[11] G. Zhang, P. Young, J. Wang, and J. Sun, Multivariable finite-time control for a 5-DOF upper-limb exoskeleton based on linear extended observers, IEEE Access, vol. 6, pp. 43213-43221, 2018.

[12] N. Thatte, J. Shah and H. Geyer, Robust and adaptive lower limb prosthetic stance control with Extended Kalman Filter-based gait phase estimation, IEEE Robotics and Automation Letters, vol. 4, no. 4, pp. 3129-3137, 2019.

[13] R. Lu, Z. Li, C.Y. Su and A. Xue, Development and learning control of a human limb with a rehabilization exokeleton, IEEE Transactions on Industrial Electronics, vol. 61, no. 7, pp 3776-3785, 2014.

[14] M.B Kang and J.H. Wa, Adaptive control of of 5-DOF upper limb exokeletaon qirh improved safety, ISA Transaaction, Elsevier vol. 52, pp. 844-852, 2013.

[15] M. Jabbari Asl, T. Nasikaya and M. Kawanishui, Neural network-based bounded control of robotic exoskeletons without velocity measurements, Control Engineering Practice, Elsevier, vol. 80, pp. 94-104, 2018.

[16] A. Riani, T. Madani, A. Benallegue and K. Djouanni, Adaptive integral terminal sliding-mode control for upper-limb rehsbilitatio exoskeleton, Control Engineering Practice, Elsevier, vol. 75, pp. 108-117, 2018

[17] J. Cao, S.Q. Xie and R. Das, MIMO sliding-mode controller for gait exoskeleton driven by pneumatic muscles, IEEE Transactions on Control Systems Technology , vol. 26, no. 1, pp. 274-281, 2018.

[18] G.P. Chen, Z.J. Du, L. He, J.Q. Wang, D.M. Wu and W. Deng, Active disturbance rejection with fast terminal sliding-mode control for a lower-limb exoskeleton in swing phase, IEEE Access, vol. 7, pp. 72343-72357, 2019.

[19] S. Mohammad, W. Huo, J. Huang H. Rifai and Y. Amirat, Nonlinear disturbance observer-based sliding-mode control of a human-driven knee-jint orthosis, Robotics and Autonomous Systems, Elsevier, vol. 75, pp. 41-49, 2016.

[20] T. Madani, B. Doachi and K. Djouani, Non-singular terminal sliding-mode controller: Application to an actuated exoskeleton, Mechatronics, Elsevier, vol. 53, pp. 136-145, 2016.

[21] N. Ajjanaramvat and M. Parnichkan, Trajectory tracking using only learning LQR with adaptive learning method of a leg-exoskeleton for disorder gait rehabilitation, Mechatronics, Elsevier, vol. 51, pp. 85-96, 2018

[22] C.A. Rodriguez, P. Ponce and A. Molina, ANFIS and MPC controllers for a reconfigurable linear limb exoskeleton, Soft Computing, Springer, vol. 21, no. 3, pp. 571-584, 2017.

[23] G.G. Rigatos and S.G. Tzafestas, Extended Kalman Filtering for Fuzzy Modelling and Multi-Sensor Fusion, Mathematical and Computer Modelling of Dynamical Systems, Taylor & Francis (2007), 13, pp. 251-266.

[24] M. Basseville and I. Nikiforov, Detection of abrupt changes: Theory and Applications, *Prentice-Hall*, 1993.

[25] G. Rigatos and Q. Zhang, Fuzzy model validation using the local statistical approach, Fuzzy Sets and Systems, Elsevier, vol. 60, no. 7, pp. 882-904,2009.

[26] G. Rigatos, Intelligent Renewable Energy Systems: Modelling and Control, Springer, 2016.

[27] G. Rigatos, P. Siano and C. Cecati, A New Non-linear H-infinity Feedback Control Approach for Three-phase Voltage Source Converters, Electric Power Components and Systems, Taylor and Francis, vol. 44, no. 3, pp. 302-312, 2015

[28] G. Rigatos, P. Siano, P. Wira and F. Profumo, Nonlinear H-infinity Feedback Control for Asynchronous Motors of Electric Trains, Journal of Intelligent Industrial Systems, Springer, 2015

[29] G. Rigatos, Modelling and control for intelligent industrial systems: adaptive algorithms in robotics and industrial engineering, Springer, 2011.

[30] G. Rigatos, Nonlinear control and filtering using differential flatness approaches: applications to electromechanical systems, Springer, 2015

[31] G.J. Toussaint, T. Basar and F. Bullo, H_∞ optimal tracking control techniques for nonlinear underactuated systems, in Proc. IEEE CDC 2000, 39th IEEE Conference on Decision and Control, Sydney Australia, 2000.

[32] L. Lublin and M. Athans, An experimental comparison of and designs for interferometer testbed, Lectures Notes in Control and Information Sciences: Feedback Control , Nonlinear Systems and Complexity, (Francis B. and Tannenbaum A., eds.), Springer, pp. 150-172, 1995.

[33] B.P. Gibbs, Advanced Kalman Filtering, Least Squares and Modelling: A practical handbook, J. Wiley, 2011.