

# Influence of the controller design on the accuracy of a forward dynamic simulation of human gait

Rosa Pàmies-Vilà<sup>a</sup>, Olga Pätkau<sup>b</sup>, Arnau Dòria-Cerezo<sup>c</sup>, Josep M. Font-Llagunes<sup>a</sup>

<sup>a</sup>*Dept. of Mechanical Engineering and Biomedical Engineering Research Centre, Universitat Politècnica de Catalunya, Diagonal 647, 08028 Barcelona, Catalonia, Spain*

<sup>b</sup>*Institute for Mechatronic Systems in Mech. Eng., Technische Universität Darmstadt, Otto-Berndt-Straße 2, 64287 Darmstadt, Germany*

<sup>c</sup>*Institute of Industrial and Control Eng., Universitat Politècnica de Catalunya, Diagonal 647, 08028 Barcelona, Catalonia, Spain*

---

## Abstract

The analysis of a captured motion can be addressed by means of forward or inverse dynamics approaches. For this purpose, a 12 segment 2D model with 14 degrees of freedom is developed and both methods are implemented using multibody dynamics techniques. The inverse dynamic analysis uses the experimentally captured motion to calculate the joint torques produced by the musculoskeletal system during the movement. This information is then used as input data for a forward dynamic analysis without any control design. This approach is able to reach the desired pattern within half cycle. In order to achieve the simulation of the complete gait cycle two different control strategies are implemented to stabilize all degrees of freedom: a proportional derivative (PD) control and a computed torque control (CTC). The selection of the control parameters is presented in this work: a kinematic perturbation is used for tuning PD gains, and pole placement techniques are used in order to determine the CTC parameters. A performance evaluation of the two controllers is done in order to quantify the accuracy of the simulated motion and the control torques needed when using one or the other control approach to track a known human walking pattern.

*Keywords:* Control Analysis, Forward Dynamics, Gait Motion

---

## 1. Introduction

Human gait dynamics has been amply studied using multibody dynamics techniques. Depending on the purpose of the study, these techniques can be used either to analyze a known motion using inverse dynamics or to simulate the motion from

joint or muscle forces through forward dynamics [1]. More precisely, the inverse dynamic analysis (IDA) is used to calculate internal joint forces and torques using acquired kinematic and kinetic data and estimated body segment parameters. On the other hand, the forward dynamic analysis (FDA) is used to obtain the motion of the musculoskeletal system as a consequence of the applied forces and torques and given initial conditions. One advantage of the FDA is that it allows the simulation or prediction of the actual behavior of the system from a given set of input actuations (at the muscle or joint level) and system parameters. Therefore, this tool might serve to anticipate, e.g., the subject's motion after a surgery or when assistive devices are used.

In an ideal case, if the results of the IDA are used as inputs of the FDA, the motion obtained through the forward simulation should match the original captured motion: as long as the inverse and forward models are the same, the results should be close to each other. However, since the forward simulation requires a numerical integration procedure, some differences appear between the captured kinematics (input of the IDA) and the simulated motion (output of the FDA). This discrepancy can be related to the integration approach and the time steps used, the interpolation schemes (needed in variable time-step algorithms), the kinematic constraint stabilization method (if it is present) or the method used to solve the differential-algebraic equations system [2]. However, the discrepancy cannot be resolved without a control system. Therefore, the use of control algorithms is crucial to ensure stability and robustness in human gait forward dynamics simulation.

In recent years, new methods for efficient control of the musculoskeletal system dynamics using optimal control methodologies have been presented [3, 4, 5]. Moreover, a growing interest in motion prediction has appeared [6, 7]. In these approaches, the basic idea is to use optimization methods to identify both force and kinematic histories based on the available information of the dynamic system. A nonlinear optimization is formulated based on the physics of the motion (dynamic equations of motion), where the objective function includes terms related to the physiology of muscle actuation and might include terms related to the aesthetics of the predicted motion as well. This function is minimized subject to some constraints; for example, dynamic equations of the musculoskeletal system, task or motion constraints, etc.

Using this type of techniques often implies a trial-and-error process, in which selecting the variables defining the motion and the drive efforts, the cost function terms (and their associated weight factors), and the appropriate physiological criteria represents a great challenge. Moreover, the use of such optimization algorithms requires several function evaluations; and, in the case of forward dynamics-based optimization, each evaluation requires the forward simulation of the complete motion.

Those techniques need an appropriate controller to stabilize the simulation and, therefore, the control approach used must be robust to perturbations and efficient in terms of computational simulation time. This paper analyzes the influence of two control strategies on the accuracy of the forward simulation of human walking, without focusing on their implementation in optimization approaches.

When the FDA of a captured motion is carried out, a dynamic inconsistency between experimental ground reaction forces and model kinematics, obtained from experimental markers, appears. Usually researchers have attempted to avoid this problem using controls on the system in order to stabilize the dynamics [8]. Those controls represent a set of non-physical forces accounting for the mentioned inconsistency, which are usually referred to as residual wrench (force and torque). This wrench is composed by linear and rotational actuators that control the absolute degrees of freedom of the models base body (pelvis or trunk in most cases).

Investigation of the real control mechanisms of muscles, that apply to reflexes or controlled motion by the central nervous system, is still a wide open subject of research in biomechanics and neurophysiology. An appropriate control to generate a forward dynamic simulation consistent with the locomotor task has not been clarified yet [2]. In the literature, there are two main approaches to face this challenging problem: following an underactuated methodology or using fully-controlled biomechanical models, in which all degrees of freedom are actuated.

The first approach is based on the principle that the human body is not a fully-actuated system, but an underactuated one. Using this methodology, the actuators can only be associated to human joints and, therefore, a control on the six degrees of freedom of the base body cannot be applied. In order to represent the foot-ground interaction researchers use a force model or a constraint methodology. This is a challenging area of research and studies following this approach employ very simple models based on passive dynamic walking to explore the natural dynamics of two-legged mechanisms (compass walker, 3-segment model, etc.) [9, 10, 11]. Another example using underactuation can be found in [12] for a jumping exercise.

In contrast, when a complex full-body model is required, authors usually propose the use of fully-controlled biomechanical models, in which all degrees of freedom are actuated. For example, the Residual Reduction Algorithm (RRA) proposed in [8] is a form of forward dynamics simulation that utilizes a controller to track model kinematics (obtained experimentally) with the aim of reducing the residual wrench (usually defined between the pelvis and the ground) to the absolute minimum that is necessary to closely follow the desired kinematics. Therefore, the external force and torque are reduced, but not eliminated, and the system is fully-actuated. Moreover, the authors of [2] proposed the use of a PD control to overcome the lack

of correlation between forward and inverse dynamic analyses. In this work, all the degrees of freedom are controlled as well, and the base segment is the pelvis. A similar approach was used in [13] with the purpose of demonstrating a computationally efficient, three-dimensional, torque actuated and forward-dynamics based model of gait, that had the potential of predicting functional outcomes of orthopedic surgeries to the musculoskeletal system. Finally, the authors of [14] proposed to combine a PD controller for each body joint together with a balanced gait controller achieved by externally manipulating the pitch of the HAT (head, arms and trunk) segment.

In the present work, the fully-controlled approach is used and the controllers driving the absolute position and orientation of the trunk are associated to the above-mentioned residual wrench. According to [14], if the model is not supported or balanced by any artificial means, poorly chosen trajectories can overwhelm the balance controller, causing the model to fall.

In the robotics field, experimental results show that the computed torque controller has very good performance characteristics and it is becoming increasingly popular [15]. However, the PD control is by far the most common control algorithm. It is amply used in controlled dynamic systems due to its simplicity in structure, robustness in operation and easy comprehension of its principle [16].

Since the musculoskeletal system is a multiple-input multiple-output highly non-linear system, the use of non-linear techniques in order to control it seems to be a good solution. For this purpose, some studies linearize the multibody system and apply a CTC algorithm [4, 17, 18]. However, for the sake of the simplicity, the PD control scheme is still used in some biomechanics studies [19, 20, 14].

This study aims to highlight how the FDA results may vary depending on the controller used, and depending on the design of this controller. For this purpose, the work presents a performance evaluation of the above control strategies for the case of human walking dynamic simulation. As far as the authors are concerned, a detailed comparison of the behavior of the two control methods in human motion simulation was not previously reported in the literature.

In order to compare the controllers, different simulation scenarios are taken into account: The study starts focusing on the numerical integration methods used and the differences obtained depending on the applied control technique. After that, the two control strategies are tested by artificially perturbing the coordinates and the external wrench (force and torque) applied to the system in a set of computational forward dynamic simulations of human walking. Differences on the simulated motion and the dynamic response of the controllers are analyzed and discussed in detail.

## 2. Dynamic Modeling

### 2.1. Biomechanical Model

A two-dimensional biomechanical human body model with 14 degrees of freedom (DOF) is used in this study. The model is a multibody system which consists of 12 rigid bodies: trunk, head, two upper arms, two forearms, two shanks, two thighs and two feet. The rigid bodies are linked with revolute joints and the model is constrained to move in the sagittal plane (Fig. 1(a)).

The subject selected to perform the experiments is a healthy adult male, 27 years old, mass 84 kg and height 1,75 m. He walks on a walkway that encloses two force plates (AMTI, Watertown, USA) sampling at 100 Hz. The motion is captured by 12 optical cameras (Natural Point, Corvallis, USA) also sampling at 100 Hz that measure the position of the 37 reflective markers attached to the subject. The anthropometric parameters together with the force plate measurements and the kinematic information are available at the library of computational multibody benchmark problems which is under development of the Technical Committee for Multibody Dynamics of the International Federation for the Promotion of Mechanism and Machine science (IFTToMM) [21].

The human body model can be defined with 13 segment points, representing the eight revolute joints combined with the end points of the five extreme segments (Fig. 1(b)). Each point is expressed using X- and Y- Cartesian coordinates (hence, 26 variables). Furthermore, 12 angular coordinates ( $\alpha_i$ ) are described: 11 relative angles ( $\alpha_1, \dots, \alpha_{11}$ ) and one absolute angle ( $\alpha_0$ ) which orientates the trunk with respect to the vertical direction in the sagittal plane (Fig. 1(c)). Therefore, the generalized coordinates vector  $\mathbf{q} \in \mathbb{R}^n$  contains  $n = 38$  configuration variables.

### 2.2. Multibody System Dynamics

The multibody system equations of motion can be expressed as:

$$\begin{cases} \mathbf{M}\ddot{\mathbf{q}} + \Phi_{\mathbf{q}}^T \boldsymbol{\lambda} & = \mathbf{Q} \\ \Phi(\mathbf{q}) & = \mathbf{0} \end{cases} \quad (1)$$

which represent a system of differential algebraic equations (DAE) with  $n$  second order ordinary differential equations (ODE) and  $m = 24$  independent algebraic constraints.  $\mathbf{M}$  is the mass matrix of the system,  $\mathbf{Q}$  is the generalized force vector,  $\boldsymbol{\lambda}$  is the vector of the Lagrange multipliers,  $\Phi$  is the kinematic constraint vector and  $\Phi_{\mathbf{q}}$  is the Jacobian constraint matrix. Finally,  $\dot{\mathbf{q}}$  and  $\ddot{\mathbf{q}}$  represent the generalized velocity and acceleration vectors, respectively.  $\Phi_{\mathbf{q}}^T \boldsymbol{\lambda}$  are the generalized forces associated with the imposed constraints.

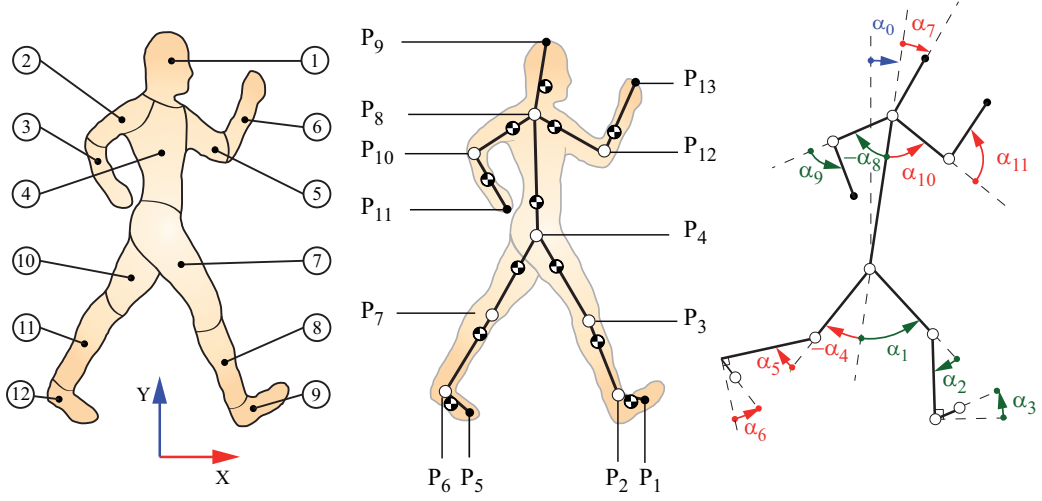


Figure 1: 2D biomechanical model of the human body.

A kinematic analysis allows us to calculate the generalized positions  $\mathbf{q}$  and their time derivatives  $\dot{\mathbf{q}}$  and  $\ddot{\mathbf{q}}$  at each time instance. In order to solve the dynamic analysis, the well known matrix-R formulation is used [22]. The equations are reduced to their state-space form, i.e., their independent coordinates  $\mathbf{z} \in \mathbb{R}^g$ . Specifically, the chosen independent coordinates  $\mathbf{z}$  are the angular variables  $(\alpha_0, \dots, \alpha_{11})$  together with the position of pelvis (also representing the hip joint), point  $P_4$  ( $x_4, y_4$ ) in Fig. 1(b). Thus, the number of independent coordinates is  $g = 14$ . Note that the independent velocities are chosen from the dependent ones:  $\dot{\mathbf{q}} = \{\dot{\mathbf{q}}^d, \dot{\mathbf{q}}^i\} = \{\dot{\mathbf{q}}^d, \dot{\mathbf{z}}\}$ , where the superscripts ‘d’ and ‘i’ indicate dependent and independent coordinates.

Using matrix-R formulation, the equations of motion can be written as [22]:

$$\mathbf{R}^T \mathbf{M} \mathbf{R} \ddot{\mathbf{z}} = \mathbf{R}^T (\mathbf{Q}_0 - \mathbf{M} \dot{\mathbf{R}} \dot{\mathbf{z}}) + \mathbf{Q}_m \quad (2)$$

where  $\mathbf{Q}_0 \in \mathbb{R}^n$  is the known generalized force vector (in this work, the gravitational forces) and  $\mathbf{Q}_m \in \mathbb{R}^g$  is the the driving forces and torques vector. Moreover, matrix  $\mathbf{R} \in \mathbb{R}^{n \times g}$  can be calculated as [22]:

$$\mathbf{R} = \begin{bmatrix} -(\Phi_{\mathbf{q}}^d)^{-1} \Phi_{\mathbf{q}}^i \\ \mathbf{I}_g \end{bmatrix} \quad (3)$$

where  $\mathbf{I}_g \in \mathbb{R}^{g \times g}$  is the identity matrix and the superscripts indicates the terms of  $\Phi_{\mathbf{q}}$  related to the dependent and independent coordinates:

$$\Phi_{\mathbf{q}} \dot{\mathbf{q}} \equiv [\Phi_{\mathbf{q}}^d | \Phi_{\mathbf{q}}^i] \begin{Bmatrix} \dot{\mathbf{q}}^d \\ \dot{\mathbf{q}}^i \end{Bmatrix} \quad (4)$$

This formulation is used to carry out both inverse and forward analyses. In an IDA the system is solved in order to obtain the driving forces and torques  $\mathbf{Q}_m = \mathbf{Q}_{IDA}$ : the analysis provides the required actuation in the form of generalized forces associated to the independent coordinates  $\mathbf{z}$ . In a FDA, motor efforts are the generalized forces computed by the IDA plus the controller dynamic contribution:  $\mathbf{Q}_m = \mathbf{Q}_{IDA} + \mathbf{Q}_C$ . Eq. (2) can be easily solved for the independent accelerations  $\ddot{\mathbf{z}}$  and the nonlinear ODE system must be numerically integrated to obtain  $\mathbf{z}_{FDA}$  and  $\dot{\mathbf{z}}_{FDA}$ .

It has to be noted that the reference or base segment of the model is the trunk, whose global position is described by 3 generalized coordinates: an absolute angle,  $\alpha_0$ , and the absolute position of the hip joint  $(x_4, y_4)$ . The other coordinates correspond to relative angles between segments  $(\alpha_1, \dots, \alpha_{11})$ . Therefore, using the explained parameterization,  $\mathbf{Q}_{IDA}$  contains an external wrench (two components of a force and one torque) applied to the trunk and 11 torques associated with joint actuation. These generalized forces do not correspond to the actual external reactions and internal joint torques, but they can be easily transformed using the procedure explained in [23]. Moreover, since  $\mathbf{Q}_C$  is the controller output vector and it is also related to the independent coordinates, it contains the external wrench and the joint torques that are needed to counteract the deviation of the controlled variable  $\mathbf{z}$  from the prescribed reference  $\mathbf{z}_{ref}$ .

Regarding ODE integration, a fixed-step solver ('ode4' from Matlab) and a variable-step one ('ode23' from Matlab) are compared. In comparison to the fixed-step solvers, the variable ones guarantee a consistent level of accuracy during all time. Sharp changes in the solution can be coped by the variable step solver by reducing the time step, where fixed-step solvers may overrun the problematic zone. About 'ode23' it has to be said that it dynamically adjusts the step size to meet the error tolerances that are specified, which is  $10^{-6}$  in our case. It solves nonstiff differential equations with the method based on Runge-Kutta(2,3) which is explained in detail in [24]. It is a one-step solver and efficient at crude tolerances and as well in the presence of mild stiffness. The difficulty of the variable-step solver in this work is that the reference data (such as reference motion and actuator torques and forces), which come from the laboratory measurements, and inverse dynamic calculations are only available in fixed time steps. Since the solver adjusts the step size dynamically during the simulation, the reference values need to be evaluated at any specific chosen time. In order to obtain continuous reference data, linear interpolation has been used. In contrast 'ode4' is a fixed-step integrator which handle the availability of the reference data in fixed time steps. It is a non-adaptive solver using as well the Runge-Kutta method of order 4.

### 3. Control System Design

As said before, two different control strategies are used: a proportional derivative (PD) controller and a computed torque control (CTC) with feedback linearization. The second order ODE system from Eq. (2) will be expressed in nonlinear state space representation. The multibody system which needs to be stabilized, is a nonlinear ‘square’ multiple input multiple output (MIMO) system (i.e., with the same number of inputs as outputs). In general, affine nonlinear systems can be expressed as:

$$\begin{cases} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) + \mathbf{g}(\mathbf{x}) \mathbf{u} \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}) \end{cases} \quad (5)$$

where, in general,  $\mathbf{u} \in \mathbb{R}^{mc}$  is the system input,  $\mathbf{y} \in \mathbb{R}^{mc}$  the system output and  $\mathbf{x} \in \mathbb{R}^{2nc}$  the state vector containing  $nc$  coordinates and  $nc$  velocities in this case [25]. The vector fields  $\mathbf{f}(\mathbf{x})$  and  $\mathbf{h}(\mathbf{x})$ , and the matrix  $\mathbf{g}(\mathbf{x})$  are nonlinear functions of the state space vector  $\mathbf{x}$  which is described as:

$$\mathbf{x} = \{x_1, \dots, x_{nc}, \dot{x}_1, \dots, \dot{x}_{nc}\}^T \quad (6)$$

For the multibody system defined previously, the state vector contains the independent coordinates  $\mathbf{z} \in \mathbb{R}^g$  as well as the independent velocities  $\dot{\mathbf{z}} \in \mathbb{R}^g$

$$\mathbf{x} = \begin{Bmatrix} \mathbf{z} \\ \dot{\mathbf{z}} \end{Bmatrix} = \{z_1, \dots, z_g, \dot{z}_1, \dots, \dot{z}_g\}^T \quad (7)$$

thus, being  $nc = g$ , The input vector  $\mathbf{u} \in \mathbb{R}^g$  contains the forces and torques that are needed to generate the reference motion.

$$\mathbf{u} = \mathbf{Q}_m = \mathbf{Q}_{IDA} + \mathbf{Q}_C \quad (8)$$

and the output vector  $\mathbf{y} \in \mathbb{R}^g$  is defined as the vector of independent coordinates  $\mathbf{z}$ . Note that, since all the DOF are controlled  $mc = nc = g = 14$ .

Using Eq. (2), the nonlinear state space representation for a second order multibody system in matrix-R formulation can be described as:

$$\begin{cases} \dot{\mathbf{x}} &= \begin{Bmatrix} \dot{\mathbf{z}} \\ \ddot{\mathbf{z}} \end{Bmatrix} = \begin{Bmatrix} \{\dot{z}_1, \dots, \dot{z}_g\}^T \\ (\mathbf{R}^T \mathbf{M} \mathbf{R})^{-1} [\mathbf{R}^T (\mathbf{Q}_0 - \mathbf{M} \mathbf{R} \dot{\mathbf{z}})] \end{Bmatrix} + \begin{bmatrix} \mathbf{0}_g \\ (\mathbf{R}^T \mathbf{M} \mathbf{R})^{-1} \end{bmatrix} \mathbf{Q}_m \\ \mathbf{y} &= \mathbf{z} = \{z_1, \dots, z_g\}^T \end{cases} \quad (9)$$



### 3.1. Proportional Derivative Control

The first control approach is a PD control. Additionally to the driving forces and torques obtained from the IDA of the reference motion, the PD control provides the necessary forces and torques to reproduce that motion in forward dynamics. The input vector with the driving forces and torques can be achieved by:

$$\mathbf{Q}_m = \mathbf{Q}_{IDA} + \mathbf{Q}_{PD} = \mathbf{Q}_{IDA} + \mathbf{K}_p \mathbf{e}_z + \mathbf{K}_d \dot{\mathbf{e}}_z \quad (10)$$

where  $\mathbf{K}_p$  is the stiffness matrix,  $\mathbf{K}_d$  is the damping matrix,  $\mathbf{e}_z = \mathbf{z}_{\text{ref}} - \mathbf{z}$  is the tracking error of the independent coordinates, and  $\dot{\mathbf{e}}_z$  is the time derivative of  $\mathbf{e}_z$ .

Large gain values in matrix  $\mathbf{K}_p$  lead to a higher response of the controller to potential disturbances. However, if this gain parameters are chosen too high, the system might oscillate and become unstable. The derivative part of the PD control (related to the gain matrix  $\mathbf{K}_d$ ) anticipates the future behavior of the tracking error since the response of the derivative component is proportional to the rate of change of the error signal [26]. The adjustable derivative time in matrix  $\mathbf{T}_d$  ( $\mathbf{K}_d = \mathbf{K}_p \mathbf{T}_d$ ) is the time interval by which the rate action advances the effect of the proportional control action [26]. Hence, the derivative gain matrix has an anticipating correcting effect. It can reduce the magnitude of overshoot, eliminates oscillations and enables the system to respond faster to external or internal disturbances. The gain matrices are chosen diagonal and have the following expressions:

$$\mathbf{K}_p = \begin{bmatrix} K_{p,1} & & 0 \\ & \ddots & \\ 0 & & K_{p,g} \end{bmatrix} \quad ; \quad \mathbf{K}_d = \mathbf{K}_p \underbrace{\begin{bmatrix} t_{d,1} & & 0 \\ & \ddots & \\ 0 & & t_{d,g} \end{bmatrix}}_{\mathbf{T}_d} = \begin{bmatrix} K_{d,1} & & 0 \\ & \ddots & \\ 0 & & K_{d,g} \end{bmatrix} \quad (11)$$

Note that the choice of null off-diagonal terms in  $\mathbf{K}_p$  and  $\mathbf{K}_d$  is for mathematical simplicity [27]. Although the presented methodology could include non-zero off-diagonal terms, as far as the authors know there is not a systematic criterion for selecting them.

In order to avoid oscillations and minimize overshoot, the gain parameters  $K_{p,i}$  and  $K_{d,i}$  have to be chosen carefully. The main problem is that the gain parameters of each controller (we have 14 controllers) cannot be easily determined since the whole dynamics of the system is strongly coupled. Additionally, the system is highly nonlinear and it does not exist, a priori, techniques for defining the gain parameters for a desired performance. For this reason, a kinematic perturbation is used for tuning PD gains and, therefore, different sets of gains are discussed in the results section.

### 3.2. Computed Torque Control

The computed torque control with feedback linearization is an approach to nonlinear control design. The main idea is to algebraically transform a nonlinear system into a linear one, so that linear control techniques can be applied (feedback linearization). In addition, due to this transformation, the new system can be controlled with  $mc$  linear independent controllers.

Roughly speaking, the CTC consists in finding an invertibility, or input-output-decoupling matrix  $\mathbf{g}(\mathbf{x})$  –where  $\mathbf{g}(\mathbf{x})$  is non-singular, or fully ranked ( $\text{rank } \mathbf{g}(\mathbf{x}) = mc$ )– and defining the input vector:

$$\mathbf{u} = \mathbf{g}^{-1}(\mathbf{x})(\mathbf{v} - \mathbf{f}(\mathbf{x})) \quad (12)$$

where  $\mathbf{v} \in \mathbb{R}^{mc}$  is the synthetic input vector and  $\mathbf{f}$  the vector containing the Lie derivative terms. Hence, the system from Eq. (5) is transformed into a decoupled set of linear equations that can be controlled using linear control techniques. It has to be mentioned that, in this case, the feedback linearization is only applied to the dynamics describing the velocity in Eq. (9). Using Eqs. (9) and (8), the derivatives of the  $mc = g$  outputs can be expressed as:

$$\underbrace{\begin{Bmatrix} \ddot{z}_1 \\ \vdots \\ \ddot{z}_g \end{Bmatrix}}_{\ddot{\mathbf{z}}} = \underbrace{(\mathbf{R}^T \mathbf{M} \mathbf{R})^{-1} [\mathbf{R}^T (\mathbf{Q}_0 - \mathbf{M} \dot{\mathbf{R}} \dot{\mathbf{z}})]}_{\tilde{\mathbf{f}}(\mathbf{x})} + \underbrace{(\mathbf{R}^T \mathbf{M} \mathbf{R})^{-1} [\mathbf{Q}_{\text{CTC}} + \mathbf{Q}_{\text{IDA}}]}_{\tilde{\mathbf{g}}(\mathbf{x})} \quad (13)$$

where  $\tilde{\mathbf{f}}(\mathbf{x})$  and  $\tilde{\mathbf{g}}(\mathbf{x})$  are the terms  $\mathbf{f}(\mathbf{x})$  and  $\mathbf{g}(\mathbf{x})$  related to the acceleration vector  $\ddot{\mathbf{z}}$  in Eq. (9). Eq. (12) can be rewritten as:

$$\mathbf{Q}_{\text{IDA}} + \mathbf{Q}_{\text{CTC}} = \mathbf{R}^T \mathbf{M} \mathbf{R} (\mathbf{v} - \tilde{\mathbf{f}}(\mathbf{x})) \quad (14)$$

Hence, using Eqs. (13) and (14) with the synthetic input vector  $\mathbf{v} \in \mathbb{R}^g$ , the nonlinear system in Eq. (9) can be transformed into a new decoupled system, which is fully linear:

$$\begin{cases} \ddot{z}_1 & = v_1 \\ \vdots & \\ \ddot{z}_g & = v_g \end{cases} \quad (15)$$

This system is a set of identical  $g$  dynamics with relative degree equal to two. Precisely, the full stability of the system is guaranteed since two times the input

vector size corresponds to the total order of the system  $2g$ , so that the stability of the internal dynamics is proved [25].

As it is shown in Eq. (15), the new linear system contains  $mc$  decoupled second order differential equations. Hence, each independent coordinate can be treated separately, and, in order to design the system's controller, it is sufficient to analyze just one equation of the system and to adopt the control adjustments of this equation to the other ones. The linearized system for one output coordinate is given by:

$$\ddot{z} = v \quad (16)$$

or in state space representation:

$$\begin{Bmatrix} \dot{z} \\ \ddot{z} \end{Bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}}_{\mathbf{A}} \begin{Bmatrix} z \\ \dot{z} \end{Bmatrix} + \underbrace{\begin{Bmatrix} 0 \\ 1 \end{Bmatrix}}_{\mathbf{B}} v \quad (17)$$

The eigenvalues of the matrix  $\mathbf{A}$  are the poles of the system, and they define the dynamic behavior in terms of stability and performance. The poles are the roots of the characteristic polynomial:

$$|(s\mathbf{I} - \mathbf{A})| = 0 \quad (18)$$

where  $\mathbf{I}$  is the identity matrix and  $s$  the Laplace operator. Eq. (18) shows that both roots are zero ( $s_1 = s_2 = 0$ ). The stability is guaranteed if  $\text{Re}(s) < 0$  for all poles. Therefore, the system is unstable without applying a control approach.

An unstable system can be stabilized by a control feedback. But before applying a suitable control strategy, the controllability of the system has to be proven. The system, expressed in Eq. (17), is completely state controllable if and only if the vectors  $\mathbf{B}$  and  $\mathbf{AB}$  are linearly independent, or what is the same the  $(2 \times 2)$  controllability matrix  $\mathbf{CO}$  is of rank 2 [26], where

$$\mathbf{CO} = [\mathbf{B}|\mathbf{AB}] = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (19)$$

Since the rank of the controllability matrix  $\mathbf{CO}$  in Eq. (19) is 2, it is proven that the unstable system in Eq. (17) is completely state controllable.

The synthetic input  $v$  is then chosen following a PD architecture [25]:

$$v = \ddot{z}_{ref} + K_d \dot{e} + K_p e \quad (20)$$

where  $K_d$  and  $K_p$  are the control parameters and the tracking error  $e$  and its time derivative  $\dot{e}$  are expressed, for each controller, as:

$$e = z_{ref} - z \quad \text{and} \quad \dot{e} = \dot{z}_{ref} - \dot{z} \quad (21)$$

with  $\ddot{e} = \ddot{z}_{ref} - \ddot{z}$ , the error dynamics can be expressed in state space representation as:

$$\begin{Bmatrix} \dot{e} \\ \ddot{e} \end{Bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ -K_p & -K_d \end{bmatrix}}_{\mathbf{A}_K} \begin{Bmatrix} e \\ \dot{e} \end{Bmatrix} \quad (22)$$

The eigenvalues of the matrix  $\mathbf{A}_K$  are the roots of the characteristic polynomial,  $|(s\mathbf{I} - \mathbf{A}_K)|$ :

$$s(s - K_d) + K_p = 0 \quad (23)$$

which are, in general, complex,  $s_{1,2} = \sigma \pm j\omega_d$ , or in terms of the control parameters,

$$s_{1,2} = -\frac{K_d}{2} \pm \sqrt{\left(\frac{K_d}{2}\right)^2 - K_p} \quad (24)$$

The stability of the dynamics is guaranteed if the real part  $\sigma$  is negative, which implies that the gains must fulfill the inequality

$$\frac{K_d}{2} > \sqrt{\left(\frac{K_d}{2}\right)^2 - K_p} \quad (25)$$

together with  $K_d > 0$ . In addition, oscillations (and consequently, overshoot) appear if the roots of Eq. (24) have imaginary part. The gain tuning consists in to setting the control gains in order to obtain a certain performance. In this paper, the following design has been considered:

1. No overshoot: the roots  $s_1$  and  $s_2$  should not contain an imaginary part, which implies

$$K_d \geq 2\sqrt{K_p} \quad (26)$$

2. Settling time: Two alternatives exist for tuning the control gains for a second order system with real poles. A first approach is to select  $K_d$  and  $K_p$  such that one real root,  $\sigma_1$ , dominates the other one,  $\sigma_2$ , with  $\sigma_2 = 5\sigma_1$ . Then, the dominant pole is related with the settling time with [25]

$$t_s \approx 4/|\sigma_1| \quad (27)$$

Alternatively, with  $K_d = 2\sqrt{K_p}$  a critical damped response is obtained and Eq. (23) has one root with multiplicity two,  $s_{1,2} = \sigma$ . The settling time for a critical damped second order systems is [25]

$$t_s \approx 5,834/|\sigma| \quad (28)$$

In the following, the approximation in Eq. (28) has been adopted (which involves  $K_d = 2\sqrt{K_p}$ ). The settling time  $t_s$  is the time that the controller will need to reach and stay within 2 % of its final value. It is worth to notice that the selection of the poles is also a compromise between rapidity of the response of the error vector and the sensitivity to disturbances and measurement noises [26]. We propose a critical damped response with  $t_s = 0,1$  s. Since the laboratory data are collected at 10ms and the gait cycle time is 1,3 s, the proposed controller will react within less than 10 % of the gait cycle. Using these conditions, the gain parameters are fixed at  $K_p = 3403$  and  $K_d = 117$ .

As explained in the beginning of this subsection, the gained information for the control design of one independent coordinate can be adopted to the other ( $mc - 1$ ) equations of the linear decoupled system in Eq. (15). Consequently,  $mc$  independent controllers using Eq. (20) have to be applied to the transformed system. In matrix formulation it can be expressed as:

$$\mathbf{v} = \ddot{\mathbf{z}}_{\text{ref}} + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} \quad (29)$$

The tracking error vector and its time derivative are the same vectors defined for the PD control. The gain matrices are also chosen diagonal as in PD control (Eq. (11)) and the determined values of  $K_p$  and  $K_d$  are used for each individual controller.

## 4. Methods

### 4.1. System Perturbations

As mentioned before, linear techniques provide tools for choosing CTC control gains according to a desired response. However, these rules are not effective for the PD control. In order to choose the proper gains for the PD controller a set of FDA are computed. The idea of these simulations is to extract some features of process dynamics and determine control parameters from the features. The FDA requires exact initial values of the independent coordinates  $\mathbf{z}_{\text{ref}}$  and independent velocities  $\dot{\mathbf{z}}_{\text{ref}}$ . What is proposed is a PD tuning based on using a wrong initial state vector. Providing wrong initial values, the controller should be able to react to the fault case and guide the motion of the system until it follows again the reference motion. The initial value of one coordinate is perturbed at each simulation:  $z_i(t = 0) \neq z_{i,\text{ref}}(t = 0)$  (14 simulations are carried out perturbing each independent coordinate). This initial error allows us to study the influence of the gain parameters on the PD controller response and, therefore, it allows us to choose the parameters according to the desired behavior.

In order to study the robustness of the controllers, another simulation scenario is investigated. The external wrench (two force components and one torque component) obtained through IDA ( $\mathbf{Q}_{\text{IDA}}$ ) is artificially perturbed at each time step. A random value is added to each wrench component according to a normal distribution with mean equal to 0 and standard deviation equal to 15%, 25% and 50% of the component actual value. This simulation takes into account an hypothetical use of a contact model where the forces can change abruptly from one instant to another. Usually, in order to reproduce the interaction between the subject and the ground, in an FDA feet are modeled by simple geometries (e.g., cylinders, spheres or ellipsoids) and a foot-ground contact model based on nonlinear spring-dampers is used [28]. The contact model in an FDA introduces uncertainties in the contact wrench and the dynamic contribution is extremely sensitive to the assumed stiffness and damping values. Therefore, it can destabilize the system. The artificial perturbation proposed aims to test this effect, since the control strategies are not aware of the external force changes acting on the system.

#### 4.2. Error Analysis

In order to compare the controllers, the root-mean-square error (RMSE) is introduced in Eq. (30). It measures the difference between a reference independent coordinate  $z_{\text{ref},i}$  and the corresponding predicted independent coordinate by means of FDA  $z_{\text{FDA},i}$ . The mean of the squared error is calculated over the total simulation time  $T$  with a total number of time steps  $K$ .

$$\text{RMSE}(z_i) = \sqrt{\frac{1}{K} \sum_{t=0}^T (z_{\text{ref},i} - z_{\text{FDA},i})^2} \quad (30)$$

where the total number of time steps is  $K = \frac{T}{\Delta t} + 1$

Finally, the normalized root-mean-square error (NRMSE) is obtained, which refers each RMSE value to the range of the independent coordinate to enable a normalized comparison.

$$\text{NRMSE}(z_i) = \frac{\text{RMSE}(z_i)}{|z_{\text{ref},i}^{\text{max}} - z_{\text{ref},i}^{\text{min}}|} \cdot 100 \quad (31)$$

Moreover, also the control torque range and the simulation time are used to compare the performance of the two control strategies. The range of the control torque is defined as the difference between maximum and minimum torque applied to the model:

$$\Delta Q_{C,z_i} = |Q_{C,z_i}^{\max} - Q_{C,z_i}^{\min}| \quad (32)$$

Note that the NRMSE and the control torque range are defined for each degree of freedom ( $i = 1, \dots, 14$ ). In some cases, and in order to have an idea of the global behavior of the system, the mean, the standard deviation and the maximum of the torque in absolute value are calculated.

## 5. Results and discussion

### 5.1. Numerical Integration

The original results of the IDA are used as input of the FDA and the motion is obtained. In a first attempt, the FDA was computed at the laboratory acquisition data frequency ( $\Delta t = 10$  ms). Because the IDA was carried out in order to determine the actuating forces and torques  $\mathbf{Q}_{\text{IDA}}$  that produce a certain reference motion  $\mathbf{z}_{\text{ref}}$ , it is expected that the results of the FDA, using these actuating forces and torques ( $\mathbf{Q}_m = \mathbf{Q}_{\text{IDA}}$ ) will ideally reach the original motion (without any controller).

However, using this integration interval, neither the fixed-step solver ('ode4') nor the variable-step one ('ode23') properly reproduce the desired movement. The multibody system became unstable and only 9% of the gait cycle could be simulated. At this point, the IDA needed to be recalculated using a smaller time step. The use of spline functions to define the motion allowed obtaining kinematic information at any instant. Specifically, a time step  $\Delta t = 1$  ms was chosen and the IDA results were again calculated.

Another FDA was computed using  $\Delta t = 1$  ms. The results obtained using 'ode4' and 'ode23' were similar, we could simulate up to 66% of the gait cycle. However, differences appeared on the simulation time: 'ode23' took 19,19 s and 'ode4' needed 3,4 s to compute the same gait cycle, i.e., the results obtained using 'ode23' took more than 5 times the simulation time used for 'ode4' integrator. In these simulations the obtained motions were really similar in both cases (no significant differences appeared on NRMSE), and therefore, the fixed-step solver was selected to perform further analyses.

Fig. 2 shows the hip joint trajectory and the lower limb joint flexion-extension angles (ankle, knee and hip joints of both legs). The reference motion is represented using a blue curve and the FDA results when the fixed-step solver is applied are plotted in green. Fig. 2 shows that during about 50 % of the cycle both curves overlap; the two motions are similar. However, after that point, the blue and green curves begin to differ and the obtained motion does not correspond to the captured gait. At 66% of the cycle the simulation crashes.

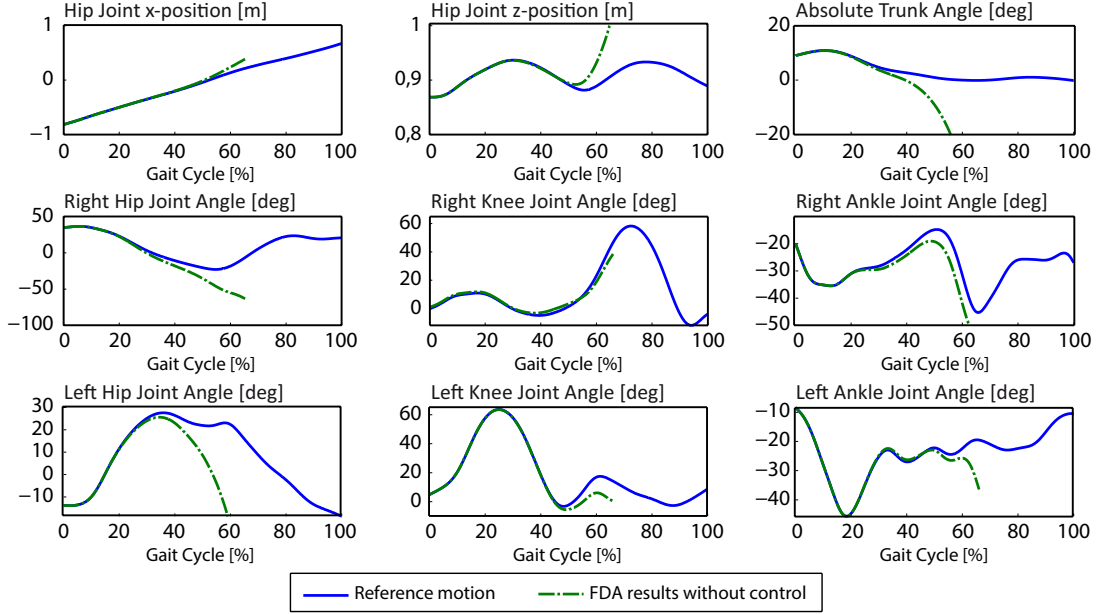


Figure 2: Evolution of trunk configuration and lower limb joint angles in the forward dynamic simulation of the gait motion without any control.

These simulations show that when the results of the inverse dynamics are used as input data for the forward dynamic analysis the obtained motion does not equal the reference one  $\mathbf{z}_{\text{ref}}$ . From now on, all the integration results are obtained using this fixed-step solver ‘ode4’, with an integration time step of 1 ms.

### 5.2. Initial condition perturbation

In order to investigate the controllers behavior a wrong initial state vector  $\mathbf{x}(t = 0)$  is used. The results presented in this section are obtained when the perturbation is applied to the right knee joint angle ( $\alpha_2$ ). Instead of the exact initial state value, this coordinate is perturbed adding half of the range of its motion.

As explained previously, the gain parameters of CTC were chosen in order to impose a settling time of 0,1 s (see section 3). However, it was not possible to perform a similar study for the PD control and several sets of PD control parameters had to be chosen. In Table 1, three selected sets of control parameter adjustments are defined.

In Fig. 3 the results obtained using the three PD controllers are plotted in black, the CTC response is plotted in red, and the reference motion is plotted in blue. The applied control torques are shown in Figs. 3(c) and 3(d).



Table 1: Performance indicators values depending on adjusted control parameters  $K_p$  and  $K_d$  of PD and CTC controllers. The wrong initial value for  $\alpha_2$  is used.

Control	Control parameters		$\Delta Q_{C,\alpha_2}$ [Nm]	NRMSE( $\alpha_2$ ) [%]	Settling time	Simulation time
	$K_p$ [ $\frac{\text{Nm}}{\text{rad}}$ ]	$K_d$ [ $\frac{\text{Nm s}}{\text{rad}}$ ]			$t_{2\%}$ [s]	$t_{\text{sim}}$ [s]
PD1	1000	10	706,74	4,57	—	7,3
PD2	100	10	65,42	10,09	0,51	7,02
PD3	100	1	85,72	8,55	—	6,9
CTC	3403	117	1021,6	6,25	0,1	4,36

It can be seen that all the controllers are able to react to the wrong initial values and guide the motion back to the desired trajectory, however the performance of each controller is different. Fig. 3(b) shows that PD1 (the controller with a higher parameter  $K_p$ ) is able to correct the initial error faster than PD2, PD3 and CTC controllers. However, as it can be seen in the same figure, neither PD1 nor PD3 are able to stabilize the motion, the error oscillates around the response and it doesn't stay within 2% of the final value: it just passes through that range on its way from one oscillation to the next. Therefore, a settling time does not exist (it is indicated using – in Table 1). It has to be said that their overshoot decrease along time and the oscillation amplitude at the end of the cycle is less than 5%. However, the defined settling time cannot be calculated for these controllers. Choosing  $K_p$  values higher than the ones in Table 1 leads to instabilities and the system performance degrades.

For the CTC controller (red curve), due to the use of the critical damped response, the controller is correcting the error without any overshoot. An NRMSE of 6,25% is obtained. Moreover, although the order of magnitude of the simulation time is the same for all the studied cases, CTC presents the faster behavior.

Regarding the control torque necessary to drive the motion, PD2 and PD3 have the lower values, 65,42 Nm and 85,72 Nm, respectively (by contrast they have the maximum NRMSE, 10,09 % and 8,55%, respectively). Figs. 3(c) and 3(d) show the the control torque applied along time –Fig. 3(d) is a zoom of the dashed box in Fig. 3(c)–. The higher values are present at the beginning of the simulation, when the controllers must reduce the largest errors. After 10% of the gait cycle all control torque magnitudes are less than 10 Nm and they tend to 0. In this study we are not concerned with the bio-physical mechanism used to provide the control torque (the muscle), we are focused on the way to virtually simulate it. Therefore, the magnitude of these torques are not used as a criterion to choose the best controller.

However, it has to be pointed out that some of these torques could not be exerted

by any biological actuator. This is for example the case of the torque needed for the CTC controller ( $\Delta Q_{C,\alpha_2} = 1021,6$  Nm, see Table 1). The maximum torque appears at the beginning of the simulation, when the controller has to correct the wrong initial value for coordinate  $\alpha_2$  see Fig. 3(c). That control torque is then reduced substantially, reaching totally affordable applied control torque values from the 10% of the gait cycle until the end of the simulation.

This study shows that if low magnitude control generalized forces were required, the PD2 controller could be considered the most adequate. However, its high NRMSE (10,09%) shows its lack of capacity to reproduce accurate human walking (see Table 1). Note that when a settling time of 0,51 s (the same as in PD2, see Table 1) is imposed to the CTC, the generalized torque is reduced to 32,1 Nm and a NRMSE of 7,43% is obtained. Therefore, when using the CTC controller, it is possible to reach the same settling time as in PD2 with a lower control torque and with more accuracy with respect to the reference motion (lower NRMSE).

The worst behavior has been detected when using PD3: a settling time cannot be calculated, the NRMSE is higher than in PD1, and the control torque range is higher than in PD2. Therefore this controller has been discarded for the rest of the study.

The behavior of the explained controllers can not be extrapolated to the remaining independent coordinates: each coordinate presents a specific performance. Fig. 4 shows the response obtained for the trunk configuration coordinates and for the lower limb joint angles. The reference motion is plotted in blue, the one obtained using PD1 and PD2 in black and the CTC curve is plotted in red.

As it can be seen in Fig. 4, the initial error only appears on the right knee joint angle. The rest of coordinates start at its reference value. However, PD controllers are not able to drive properly the prescribed motion of some coordinates: the error on the vertical coordinate of the hip joint increases along time; neither PD1 nor PD2 are able to follow the desired motion of the absolute angle of the trunk; moreover, some errors appear at the beginning of the cycle for the right and left hip joint angles. For the PD controllers, this effect is related to the coupled system dynamics: an error on a single coordinate effects all the system, moreover, the gains cannot be separately tuned, since a change in  $K_{p,i}$  or  $K_{d,i}$  affects the behavior of all the controllers.

In contrast, very accurate dynamic results can be achieved when the CTC is used: the coordinates follow the reference motion as desired (red curve in Fig. 4). An error on the knee coordinate has no effect on the other coordinates.

The performance indicators used above can be calculated for each independent coordinate. For the sake of simplicity (note that there are 14 actuators) the mean NRMSE, its standard deviation (SD) and the maximum value obtained are sum-

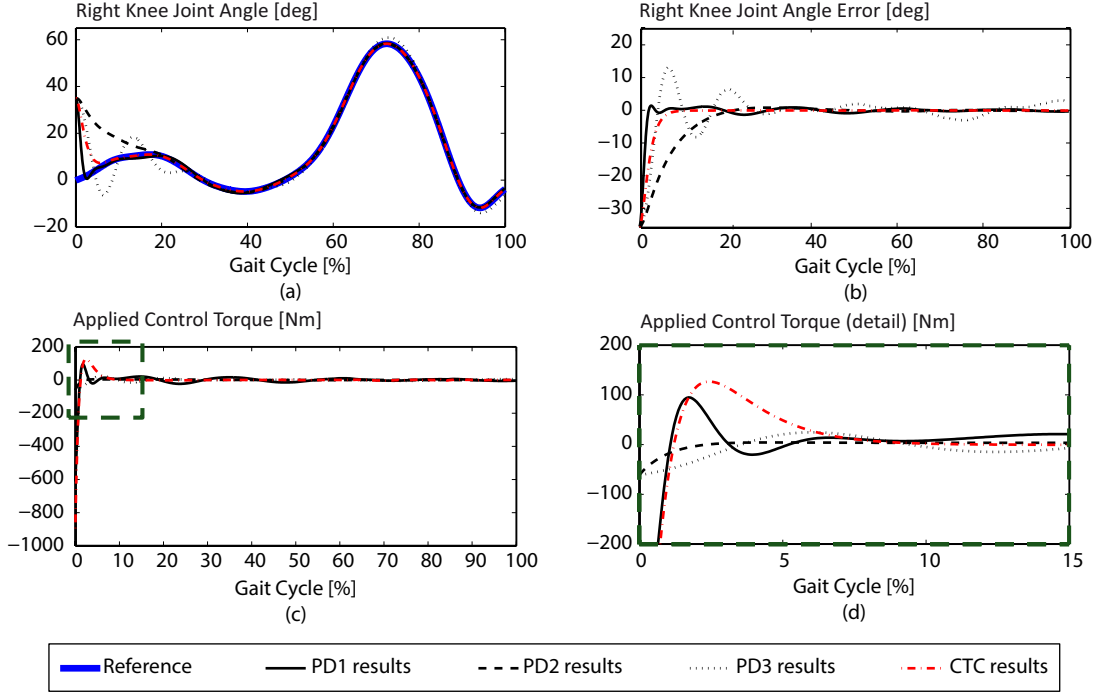


Figure 3: Evolution of the right knee joint angle ( $\alpha_2$ ) and applied control torque related to this coordinate. A wrong initial condition for  $\alpha_2$  is used. (a) Joint angle evolution, (b) Absolute joint angle error, (c) and (d) Applied control torque

marized in Table 2. Moreover also the mean of the controller torques, its standard deviation (SD) and the maximum value obtained are shown in this table.

The dynamic effect of the controllers are force acting on the hip joint (expressed in N) and a set of torques acting between segments (expressed in Nm). In order to be consistent and not to mix magnitudes, the values in Table 2 just take into account the 12 angular controllers (they present indeed the maximum error values).

As expected, comparing the three controllers it can be seen that the CTC shows all around the best performance: it is the controller with the lowest NRMSEs in terms of mean, SD, and maximum values; and, moreover, the CTC mean NRMSE is one order of magnitude lower than the one of PD controllers. Only CTC can drive properly the motion with an mean NRMSE of 0,57% with the desired settling time of 0,1 s. The maximum NRMSE of the CTC (6,25%) is the same value as the NRMSE of Table 1: the maximum error appears on the knee ankle joint (where the error occurs). However, the maximum NRMSEs of PD controllers are higher than the NRMSE ( $\alpha_2$ ) in Table 1: the largest error appears on a non-perturbed coordinate.

PD2 presents a lower control torque range than PD1 (in mean and also in SD

Table 2: Global comparison of both control strategies -PD and CTC- with wrong initial value at  $\alpha_2$ . Statistical results obtained for the 12 angular controllers.

Control	Mean NRMSE [%]	SD NRMSE [%]	Max NRMSE [%]	Mean $\Delta Q_C$ [Nm]	SD $\Delta Q_C$ [Nm]	Max $\Delta Q_C$ [Nm]
PD1	3,02	3,79	14,21	135,26	186,85	706,74
PD2	3,51	5,11	18,24	11,01	17,27	65,42
CTC	0,57	1,66	6,25	434,45	758,29	2054,1

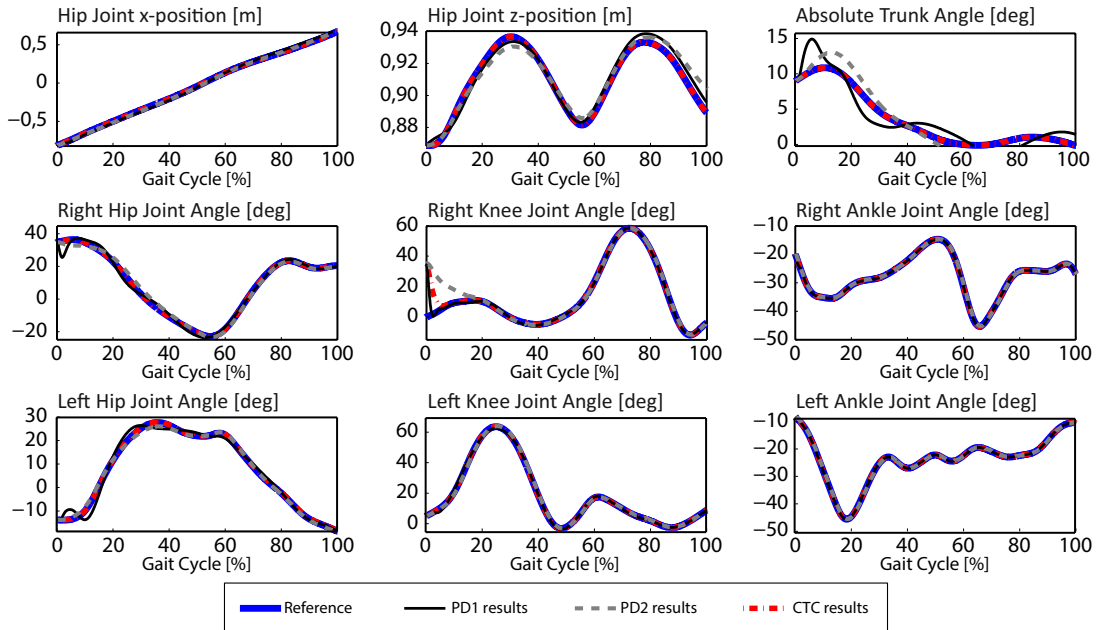


Figure 4: Evolution of trunk configuration and lower limb joint angles. Comparison of both control strategies -PD and CTC- with wrong initial value at  $\alpha_2$

value). Its maximum control torque is one order of magnitude less than PD2 and two orders of magnitude less than CTC. However, both PD controllers are not able to correct the error of some coordinates: they do not stay within 2% of the final value and there is no settling time for some coordinates (vertical component of hip joint, absolute trunk angle and hip joint angles). In contrast, PD1 features a lower mean, SD and maximum NRMSE value.

Since the tracking error behaves as a critically damped second-order system, it seems reasonable to assume that CTC will always outperform a simple PD algorithm. However, note that CTC technique requires the knowledge of many physiological parameters (for the feedback linearization process), but the PD controller does not. So, in this study it was also interesting to investigate how a wrong (but reasonable) estimation of physiological parameters affected both controllers. Under this condition, CTC still performed better than the PD controller, which was not known *a priori*.

This initial perturbation test has been used in order to tune the PD gain parameters in the sense that different gains has been imposed and analyzed. The results show that PD1 is neither better nor worse than PD2. The use of one or other will depend on the application. Therefore, in the next section both controllers will still be used and compared.

### 5.3. Control Behavior in non-perturbed FDA of Human Gait

Using the above controllers (PD1, PD2 and CTC) a new FDA is carried out. In this new case, the wrong initial condition is not applied. Since no uncertainties are present, it is expected that the controllers will be able to properly drive the system. However, significant errors appear on the vertical coordinate of the hip joint and on the absolute trunk angle coordinate when PD controllers are used. Fig. 5 shows the results for these two specific coordinates. Black curves represent the PD controllers, the red one is related to CTC and the reference motion is plotted in blue. Although, the biomechanical model is able to follow the prescribed motion, PD1 and PD2 cannot drive accurately these coordinates. In contrast, when CTC is applied, the controlled variables are concurrent with the reference variables.

As expected the error in CTC remains close to 0, but what is interesting to see is that the PD controllers the error increases along the gait cycle, and no settling time can be calculated. Regarding the vertical component of the hip joint, PD1 presents an NRMSE of 3,08% and a  $\Delta Q_C$  of 4,27 N. Instead, the NRMSE when PD2 is used is 6,37% and the  $\Delta Q_C = 1,3$  N. For the absolute trunk angle, the NRMSE of PD1 is 2,92% and it requires a  $\Delta Q_C$  of 18,87 Nm. In contrast, PD2 presents a higher NRMSE (8,45 %) and a lower  $\Delta Q_C$  (3,02 Nm).

As in the previous subsection, in order to evaluate the global behavior of the

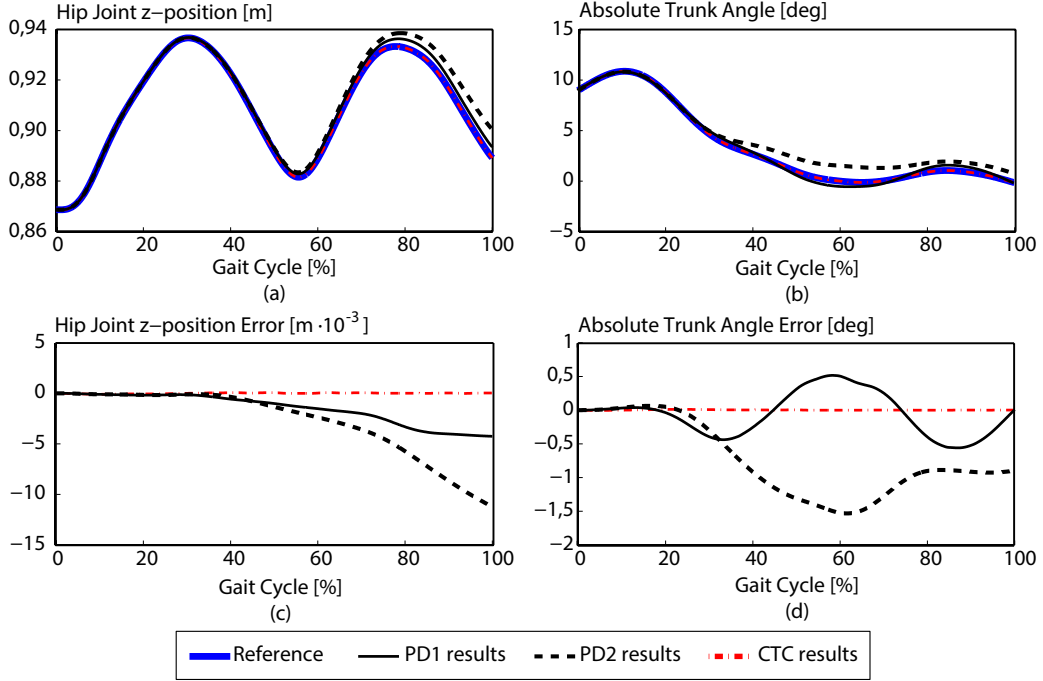


Figure 5: Kinematic results for the vertical coordinate of the hip joint (a) and (c); and for the absolute trunk angle (b) and (d). Results obtained without applying any perturbation.

system, the mean, SD and maximum values of the NRMSEs and the control torque range are calculated (see Table 3). The maximum NRMSE for the PD controllers shown in Table 3 corresponds to the absolute trunk angle. The CTC control approach features again a better performance than the two PD controllers since its mean NRMSE is one magnitude order lower than the best PD control (PD1). Regarding the control torque range, CTC applies higher torques. The simulation time is similar for all three cases.

As seen in Fig. 5, neither PD1 nor PD2 are able to exactly reproduce the considered coordinates. However, in general terms, the behavior of PD1 seems to be more appropriate than PD2: Its NRMSEs are lower than in PD2 results (see Table 3). In contrast, higher control torques and forces must be applied.

Note that the obtained performance indicators in this new simulation are, in general, one order of magnitude lower than the results for the previous case (see Table 2), and they are two orders of magnitude lower for the CTC. In this second case, no external perturbations are added to the system state and, therefore, the controllers only need to react to the unavoidable simulation divergences between FDA and IDA.

Table 3: Global comparison of both control strategies -PD and CTC- without any applied perturbation. Statistical results obtained for the 12 angular controllers.

Control	Mean NRMSE [%]	SD NRMSE [%]	Max NRMSE [%]	Mean $\Delta Q_C$ [Nm]	SD $\Delta Q_C$ [Nm]	Max $\Delta Q_C$ [Nm]	Simulation time [s]
PD1	0,38	0,75	2,91	5,31	5,24	18,87	7,76
PD2	1,47	2,16	8,44	1,91	1,47	5,45	7,3
CTC	0,04	0,02	0,07	7,71	8,61	27,51	6,59

#### 5.4. External wrench perturbation

The external wrench, obtained from the IDA ( $\mathbf{Q}_{IDA}$ ), is perturbed and used to test the capacity of the controllers to keep the human motion stable against external disturbances. Specifically, three different forward dynamic simulations are carried out for each considered controller (PD1, PD2 and CTC). In those simulations, the external wrench (two force components and one torque component) related to the body-environment interaction is modified according to a normal distribution with 0-mean and standard deviation equal to 15%, 25% and 50%, respectively, of the component actual value (Error SD in Table 4).

Table 4 shows the obtained values. PD1 can drive the motion and the obtained errors are similar to the other studied cases. However, PD2 control is not able to follow the prescribed motion. For these three simulations the NRMSEs are greater than 25%. Using PD2 the obtained motion significantly differs from the original one. It must be highlighted that since the  $K_p$  values of PD1 are higher than those of PD2, the control torque applied by PD1 can also be higher. Therefore, PD1 is able to react faster to these disturbances and therefore it performs better than PD2.

The fundamental difficulty of using the designed PD controllers is that the control system has no direct knowledge of the process and the feedback signals are multiplied with constant parameters. This control strategy has difficulties in the presence of non-linearities (changing process behavior) and has a delay responding to large disturbances. Something that becomes evident in these last simulations for PD2. Moreover, since the control parameter gain matrices are chosen diagonal, these controllers do not take possible coupling among DOF into account.

Regarding the CTC, this controller drives properly all the coordinates. In the worst case, the mean NRMSE is less than 0,25%, so the obtained motion is close to the original one. Note that the CTC control approach and therefore, the implementation of the feedback linearization, is not aware of any uncertainties applied to the model parameters and does not consider any disturbances acting on the plant. The system

Table 4: Global comparison of both control strategies -PD and CTC- when the IDA external wrench is perturbed. Statistical results obtained for the 12 angular controllers.

<b>Error</b>	<b>Mean</b>	<b>SD</b>	<b>Max</b>	<b>Mean</b>	<b>SD</b>	<b>Max</b>	<b>Simulation</b>
<b>SD</b>	<b>NRMSE</b>	<b>NRMSE</b>	<b>NRMSE</b>	$\Delta Q_C$	$\Delta Q_C$	$\Delta Q_C$	<b>time</b>
[%]	[%]	[%]	[%]	[Nm]	[Nm]	[Nm]	[s]
<b>PD1</b>							
15	0,33	0,56	2,2	5,57	5,1	16,04	5,22
25	0,41	0,72	2,82	6,92	6,96	23,32	5,13
50	0,78	1,61	6,20	11,39	13,68	49,39	4,75
<b>PD2</b>							
15	29,61	107,04	401,52	6,76	16,32	63,32	5,22
25	48,95	178,29	668,39	10,26	27,36	104,9	5,14
50	98,67	357,42	1340,37	19,84	54,18	210,90	5,14
<b>CTC</b>							
15	0,08	0,08	0,33	34,7	74,74	271,36	5,25
25	0,12	0,13	0,54	40,4	83,31	276,51	5,01
50	0,23	0,27	1,08	56,27	115,671	346,16	4,93

behavior when a random perturbation is applied to the system is not known *a priori*. However, CTC algorithm is able to follow the prescribed motion even with significant external errors. Logically the control torque range increases with higher errors since it has to correct larger disturbances. The IDA torque fault case involves, in addition, high angular acceleration errors.

This test allows us to study the capacity of the controllers to restore and maintain upright posture against external perturbations, and therefore, knowing in what manner stable gait patterns are recovered in response to external perturbations. The results prove the robust performance of the CTC control against the PD counterpart. The CTC control approach is highly robust against dynamic uncertainties and also to kinematic errors. In contrast, the PD behavior is very dependent on the gain parameters and could not be able to follow a reference in presence of high uncertainties.

Finally, neither the percentage of error on the external wrench nor the type of control used to drive the motion have an effect on the final simulation time.

## 6. Conclusions

In this work, inverse and forward dynamic approaches have been implemented for a human gait analysis using a 12 segment whole-body 2D model with 14 DOF. The IDA is computed with the aim of obtaining a set of forces and torques dy-



namically consistent with the captured motion without adding any residual reaction. These forces and torques are then applied to the multibody system when the FDA is computed.

Different numerical integration methods are introduced in order to carry out the forward dynamic simulations ('ode4' and 'ode23' from Matlab). In this study the kinematic results obtained for both integrators were really similar (no differences appear on NRMSE). Comparing their performances, the fixed-step solver 'ode4' from Matlab performed 5 times faster than the variable one ('ode23'). Hence, the further forward dynamic simulations have been carried out using the fixed-step integration solver 'ode4'.

Using those input forces and torques without any control the simulation drift appears at around 50% of the gait cycle. Due to numerical errors in the integration process and the unstable character of the human motion, control techniques must be applied. In order to see the effect of the choice of the controller on the accuracy of the simulated motion and the control torques needed during the forward dynamic simulation, two different control approaches are introduced and designed: a proportional derivative control and a computed torque control using feedback linearization. The gain parameters have been investigated and they have been chosen according to the control response. Three different sets of  $K_{p,i}$  and  $K_{d,i}$  have been used for the PD controller. NRMSE and settling time have been used to compare the controllers behaviors.

The main advantage of the PD control is that it is very simple to implement and involves very few computations compared to the nonlinear dynamic equations. However, the 14 controllers are strongly coupled: a change in the gain parameters of one controller effects the whole dynamics of the system. In contrast, one of the most important advantages of the CTC is that the obtained multibody system is fully linearized. Therefore, linear control techniques can be applied. Moreover, the desired closed loop pole locations determine the speed and the damping of the response. In particular, the settling time can be set in order to dictate the response of the control system.

When a wrong initial position is used in order to determine the controller gains, the study of the behavior of the knee joint shows that neither PD1 nor PD2 are able to stabilize the motion with a settling time within one gait cycle. They present oscillations higher than 2% of its final desired value. In contrast, when CTC is used, very accurate dynamic results are achieved with a settling time of 0,1s although bigger control torques are needed in this case. However, note that the aim of the paper is not to obtain physiologically realistic torques, but a robust and computationally efficient control system able to stabilize the forward dynamic simulation of human gait.

Applying no errors to the system, although all the control approaches are able to follow the reference motion, some important differences appear in some coordinates when the PD controllers are used. The vertical hip joint position and the absolute trunk angle present higher errors than the other coordinates. These errors cannot be reduced increasing gain parameters. The dynamics of the system is highly coupled and a change in one parameter affects the behavior of the other coordinates. In contrast, CTC control performs without overshoot, faster and more accurately with a factor of 10 as compared with the PD control.

The PD control approach has its limitations in the case of external wrench perturbations. The results of the forward dynamic simulation fails depending on the gains used. The CTC control, instead, is able to guide the human motion during the whole simulation following the reference data with a highly accurate response.

This work points out that, although there are a lot of papers in the literature using a PID controller for forward dynamic analysis of human gait, this is not the best approach as seen from the results obtained in the different simulation scenarios. In summary, it can be pointed out that the CTC control with feedback linearization is a powerful control system for nonlinear multibody systems. The results show that it is a tool with high potential for human motion control since its performance is very robust. Authors suggest that this tool might have a high potential in forward dynamics-based algorithms for human motion prediction. In contrast, the use of PD control could fail to restore and maintain upright posture under certain conditions.

## 7. References

- [1] Ambrósio J. A. C. and Kecskeméthy A. (2007) *Multibody Dynamics of Biomechanical Models for Human Motion via Optimization*, Computational Methods in Applied Sciences, 4, 245-272.
- [2] Quental C., Folgado J., Ambrósio J. and Silva M.T. (2015) *A simple controller to overcome the lack of correlation between forward and inverse dynamic analysis of human motion tasks*, Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics October 8.
- [3] Ackermann M. and Van den Bogert, A.J. (2010) *Optimality principles for model-based prediction of human gait*, J Biomech, 43(6), 1055-1060.
- [4] Thelen D.G. and Anderson F.C (2006) *Using computed muscle control to generate forward dynamic simulations of human walking from experimental data*, Journal of Biomechanics 39 (6), 1107-1115.

- [5] Van den Bogert A.J., Blana D. and Heinrich D. (2011) *Implicit Methods for Efficient Musculoskeletal Simulation and Optimal Control*, IUTAM Symposium on Human Body Dynamics, Procedia IUTAM, 2, 297-316.
- [6] Fregly B.J., Reinbolt J.A., Rooney K.L., Mitchell K.H. and Chmielewski T. L. (2007) *Design of Patient-Specific Gait Modifications for Knee Osteoarthritis Rehabilitation*, IEEE Trans. Biomedical Engineering, 54(9), 1687-1695.
- [7] Kim J.H. Abdel-Malek, Yang J. and Marler R.T. (2006) *Prediction and Analysis of Human Motion Dynamics Performing Various Tasks*, International Journal of Human Factors Modelling and Simulation, 1, 69-64.
- [8] Delp S.L., Anderson F.C., Arnold A.S., Loan P., Habib A., John C.T., Guendelman E., Thelen D.G. (2007) *OpenSim: Open-source Software to Create and Analyze Dynamic Simulations of Movement*, IEEE Transactions on Biomedical Engineering, 54(11), 1940-1950.
- [9] Asano F. (2015) *Stability analysis of underactuated compass gait based on linearization of motion*, Multibody System Dynamics, 33(1), 93111.
- [10] Goswami A., Thuilot B., Espiau B.(1998) *A study of the passive gait of a compass-like biped robot: symmetry and chaos*, The International Journal of Robotics Research December, 17 (12), 12821301.
- [11] Grizzle J.W., Abba G., Plestan F.(2001) *Asymptotically stable walking for biped robots: analysis via systems with impulse effects*, IEEE Transactions on Automatic Control, 46(1), 5164.
- [12] Seth A., Pandy M.G.(2007) *A neuromusculoskeletal tracking method for estimating individual muscle forces in human movement*, Journal of Biomechanics, 40, 356366.
- [13] Fluit R., Van der Krogt M.M., Van der Kooij H., Verdonschot N., Koopman H.F.J.M. (2012) *A simple controller for the prediction of three-dimensional gait*, Journal of Biomechanics 45, 26102617.
- [14] Millard M., McPhee J., Kubica E., (2008) *Multi-step forward dynamic gait simulation*, Bottasso C.L. Editor, Multibody Dynamics: Computational Methods in Applied Sciences, 12, 2543.

- [15] Bolek W., Sasiadek J. and Wisniewski T. *Linearization of nonlinear MIMO model of large power plant station*, American Control Conference, 2000. Proceedings of the 2000 (6), 4435-4436.
- [16] Ghafari A.S., Meghdari A. and Vossoughi G.R. (2009) *Forward dynamics simulation of human walking employing an iterative feedback tuning approach*, Journal of Systems and Control Engineering, 223(3), 289-297.
- [17] Ünver N.F., Tümer S.T. and Ozgören M.K. (2000) *Simulation of human gait using computed torque control*, Technology and Health Care, 8(1), 53-66.
- [18] Thelen D.G., Anderson F.C and Delp S.L. (2003) *Generating forward dynamic simulations of movement using computed muscle control*, J Biomech 36,321-328.
- [19] Masani K., Vette A.H. and Popovic M.R. (2006) *Controlling balance during quiet standing: proportional and derivative controller generates preceding motor command to body sway position observed in experiments*, Gait and Posture, 23, 164-172.
- [20] Kuriyama S., Kurihara Y., Irino Y. and Kaneko T. (2002) *Physiological gait controls with a neural pattern generator*, J. Visual. Comp. Animat., 13, 107119.
- [21] Pàmies-Vilà R., Font-Llagunes J.M., Lugrís U, Alonso F.J. and Cuadrado J. (2014) *A Computational Benchmark for 2D Gait Analysis Problems*, EU-COMES 2014. New Trends in Mechanism and Machine Science, pp. 505-514, Springer, 2015. Guimaraes, Portugal, 2014-09.
- [22] García De Jalón J. and Bayo E. *Kinematic and Dynamic Simulation of Multibody Systems*, Springer-Verlag, 1994, New York, USA.
- [23] Lugrís U., Carlin J., Pàmies-Vilà R., Font-Llagunes J.M. and Cuadrado J. (2013) *Solution methods for the double support indeterminacy in human gait*, Multi-body System Dynamics, 30(3), 247-263.
- [24] Bogacki P. and Shampine L.F. (1989) *A 3(2) pair of Runge - Kutta formulas*, Applied Mathematics Letters 4 (2), 321-325.
- [25] Slotine J.J.E. and Li W. *Applied nonlinear control*, Prentice-Hall Englewood Cliffs, New Jersey, 1991.
- [26] Ogata K. and Yanjuan J. *Modern Control Engineering*, Prentice-Hall Englewood Cliffs, 1970, New Jersey, USA.

- [27] Haddadin S. *Towards Safe Robots. Approaching Asimovs 1st Law*, Springer Tracts in Advanced Robotics 90, Springer, 2014, Germany.
- [28] Pàmies-Vilà R., Font-Llagunes J.M., Lúgrís U. and Cuadrado J. (2014) *Parameter identification method for a three-dimensional foot-ground contact model*, Mechanism and Machine Theory 75, 107-116.