# Comparison of several muscle modeling alternatives for computationally intensive algorithms in human motion dynamics

M. Lamas[1] · F. Mouzo[1] (ORCID) · F. Michaud[1] · U. Lugris[1] · J. Cuadrado[1]

## Abstract

Several approaches are currently employed to address the predictive simulation of human motion, having in common their high computational demand. Muscle modeling seems to be an essential ingredient to provide human likeness to the obtained movements, at least for some activities, but it increases even more the computational load. This paper studies the efficiency and accuracy yielded by several alternatives of muscle modeling in the forward-dynamics analysis of captured motions, as a method that encompasses the computationally intensive character of predictive simulation algorithms with a known resulting motion which simplifies the comparisons. Four muscle models, the number of muscles, muscle torque generators, muscular synergies, and look-up tables for musculotendon lengths and moment arms are considered and analyzed, seeking to provide criteria on how to include the muscular component in human multibody models so that its effect on the resulting motion is captured while keeping a reasonable computational cost. Gait and vertical jump are considered as examples of slow- and fast-dynamics motions. Results suggest that: (i) the rigid-tendon model with activation dynamics offers a good balance between accuracy and efficiency, especially for short-tendon muscles; (ii) including muscles in the model leads to a decrease in efficiency which is highly dependent on the muscle model employed and the number of muscles considered; (iii) muscle torque generators keep the efficiency of skeletal models; (iv) muscular synergies offer almost no advantage for this problem; and (v) look-up tables for configuration-dependent kinematic magnitudes have a non-negligible impact on the efficiency, especially for simplified muscle models.

**Keywords** Muscle modeling · Muscle torque generator · Synergies · Gait · Vertical jump · Biomechanics · Efficiency

## 1 Introduction

Predictive simulation of human motion is a topic of great interest today [1, 2], as it opens the way to applications with high economic and societal impact, as anticipation of surgery outcome, orthotic-prosthetic design and customization, control of humanoid robots and virtual

---

✉ F. Mouzo

1   Lab. of Mechanical Eng., University of La Coruña, Mendizabal s/n, 15403 Ferrol, Spain

characters, preevaluation of training methods in sport and manual task procedures in labor environments, etc. Robotics, biomechanics, multibody dynamics, and computer graphics are some of the communities that are actively investigating in this field.

In some cases, the objective is to design a motion that minimizes or maximizes some measure of performance while satisfying some constraints, what has been called trajectory optimization. Optimal control is the most popular and established approach for this purpose [3–10]. In other cases, the objective is to create controllers that govern the motion, either by mimicking the behavior of the neural system [11–15] or by means of artificial intelligence (AI) based algorithms [16, 17].

Predictive simulation algorithms can be stated in direct or inverse form, so that they can require either the forward integration of the system differential equations or just the solution of an algebraic set of equations. But, in any case, they are always iterative and highly computationally intensive [1].

Although predictive simulation of human motion can be based on just skeletal models, there is a general consensus that inclusion of the muscular component is relevant to capture the features of real human movement in some activities, i.e., to obtain human-like motions, and to get insight into other aspects, as tendon behavior or energy consumption [18, 19]. However, using musculoskeletal models further increases the computational load of the resulting algorithms [2], so that a muscular modeling must be sought that is efficient enough to keep the execution times within reasonable limits (specially for applications that are expected to work in real time, but not only) while being capable of capturing the key features of human movement to provide human-like solutions.

Therefore, the study of efficiency and accuracy of different muscular modeling alternatives can help provide criteria on how to build the musculoskeletal models required by algorithms for predictive simulation of human motion. The alternatives studied in this paper are the following: (i) muscle models, including the full Hill model, the Hill model with rigid tendon and activation dynamics, the Hill model with rigid tendon and without activation dynamics, and a nonphysiological model; (ii) the number of muscles; (iii) muscle torque generators (MTGs); (iv) muscular synergies; as well as (v) look-up tables for muscular lengths and articular moment arms.

The literature gathers contributions that have addressed some of these alternatives. For example, in [20], the full Hill model of the muscle, called the equilibrium musculotendon model, was compared with two variants: the damped equilibrium musculotendon model, which included a damper in parallel with the active contractile element and the passive elastic element; and the rigid-tendon musculotendon model, which treated the tendon as inextensible. Benchmarks with one single muscle were proposed for maximal and submaximal activations, and for muscles having long and short tendons, concluding that the damped and rigid-tendon models were suitable for the long- and short-tendon cases, respectively, providing more efficiency than the equilibrium model, with an acceptable error. The models were also used to replicate a couple of real experiments, yielding satisfactory results, and were finally compared when being used in gait, confirming the improvement in efficiency provided by the damped- and rigid-tendon models. The effect on the efficiency of the number of muscles considered in the model was addressed in [18], where a simplified model with nine muscles per leg and a complex model with 43 muscles per leg were compared within an optimal control algorithm for muscle dynamics computation in a prescribed walking motion, showing that the simple model was almost two orders of magnitude faster on average than its complex counterpart. MTGs were used in [9], implementing this type of muscle modeling within optimal control algorithms that predicted the executions of three sport tasks; although an explicit comparison of efficiency with the case of using

conventional Hill-type muscles was not carried out, the obtained computation times were clearly lower than those required by other authors for similar problems, in which the MTG simplification had not been made. The efficiency and accuracy (measured with respect to EMG recordings) obtained, for the muscle recruitment problem in gait, by several muscle models and by a synergy-based approach, solved in all cases through inverse-dynamics based optimization were compared in [21]; as it happened in [20], it was found that the rigid-tendon muscle model offered a good compromise between efficiency and accuracy but, on the other hand, it was also showed that the best results at both levels were obtained with a nonphysiological muscle model; the use of muscular synergies neither improved efficiency nor accuracy. Finally, [22] provided a method based on polynomials to approximate musculotendon lengths and moment arms, and reported notable gains in efficiency when computing musculoskeletal kinematics with respect to previously proposed methods.

A relevant point is how to conduct the study, i.e., how to compare the efficiency and accuracy of the different alternatives mentioned above in a way that is meaningful and allows to extract general conclusions for the use of such muscular modeling alternatives in algorithms for the predictive simulation of human motion, without falling in the particularity of using a specific algorithm, and while avoiding the high level of complexity and, in many cases, the low robustness of the algorithms.

The approach adopted here was to test the abovementioned alternatives of muscular modeling in the forward-dynamics analysis of captured real motions. A controller tracked the joint trajectories while the measured foot-ground reactions were applied to the feet. This approach, at skeletal level only, was described in [23]. Its extension to musculoskeletal level is addressed in this paper. In this way, the iterative and computationally demanding nature of predictive simulation algorithms is gathered, but their complexity is avoided. As it could be expected that motion intensity affects the results, two activities were selected for the study: gait, as representative of slow-dynamics motions, and vertical jump, as representative of fast-dynamics motions.

The contributions of this paper are the following: (i) the systematic comparison of several muscle modeling alternatives (four muscle models, the number of muscles, muscle torque generators, muscular synergies and look-up tables for configuration-dependent kinematic magnitudes) within a framework that, on the one hand, shows the computationally intensive character of predictive simulation algorithms and, on the other hand, allows for a fair comparison of efficiency and accuracy as the resulting motion is given; (ii) the proposed approach for the forward-dynamics analysis of captured human motion, which does not require the unified integration of the multibody and muscle dynamics equations; (iii) the recommendations on muscular modeling extracted as conclusions of the work.

The remainder of the paper is organized as follows. Section 2 describes the approach that was adopted to conduct the comparison of the different alternatives for muscular modeling, the experiments carried out and the subjects that participated in them, and the human multibody model employed at skeletal and musculoskeletal levels. Section 3 is devoted to list and explain the four muscle models that were compared and how the force-sharing problem was addressed. Section 4 is focused on the MTGs, Sect. 5 deals with muscular synergies, and Sect. 6 shows the interpolation methods considered for musculotendon lengths and moment arms. Section 7 presents the obtained results and discusses them and, finally, Sect. 8 gathers the conclusions of the work.

**Fig. 1** Cointegration method for the forward-dynamics analysis of captured motions. The coordinates defining the motion of the multibody system and their derivatives are vectors $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$. The Jacobian matrix containing the moment arms is $\mathbf{J}_m$, the vector of joint torques is $\mathbf{Q}_{CTC}$, the vectors gathering the upper and lower limits of the muscular forces are $\mathbf{F}_{min}$ and $\mathbf{F}_{max}$, and the vector of muscular forces obtained after solving the force-sharing problem is $\mathbf{F}_{opt}$. The vectors of muscular excitations and activations are, respectively, $\mathbf{u}$ and $\mathbf{a}$. Superscripts $n$ and $n+1$ denote the time step. Subscript $d$ denotes the desired (captured) motion. Functions inside shaded boxes may be necessary or not depending on the muscle model adopted
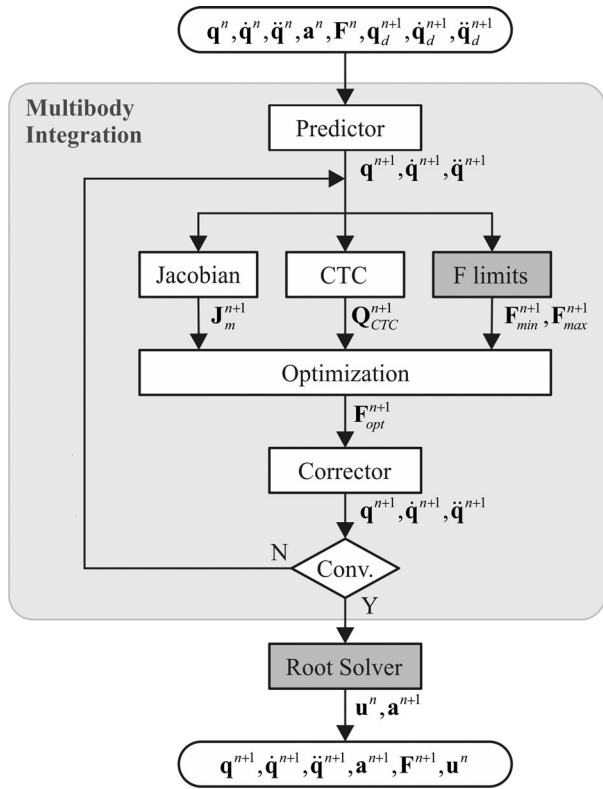


## 2 Problem approach, subjects, experiments, and multibody models

As said in the introduction, the several alternatives of muscle modeling were tested in the context of the forward-dynamics analysis of captured real motions, an approach that possesses the iterative and computationally demanding characteristics of the algorithms used for predictive simulation, but considers a known resulting motion, thus making easier to compare the different muscle modeling alternatives. The method is similar to the popular Computed Muscle Control (CMC) method [24]. It is based on a cointegration scheme so that, unlike CMC, the unified integration of the multibody and muscle dynamics equations is not required, thus allowing the use of any multibody code for generating and integrating the multibody dynamics equations, while the muscle dynamics can be simulated within a different framework.

As it can be seen in Fig. 1, the multibody equations are integrated using an implicit integrator, in a predictor–corrector scheme that combines the equations of motion and the integrator equations to yield a nonlinear system of equations with the configuration-level coordinates as primary variables, which is solved by a Newton–Raphson procedure. In this case, the trapezoidal rule was chosen as integrator with a time-step size of 10 ms, and the equations of motion of the multibody system were derived by means of a semirecursive formulation in joint coordinates [25].

Once inside a certain iteration, in the predictor block the coordinates at position level for the next time point are extrapolated from the current positions, velocities, and accelerations, by means of the constant-acceleration kinematic formula. Then, the joint torques required to

**Fig. 2** The selected movements and the subjects who performed them: (**left**) gait; (**right**) vertical jump

track the captured joint trajectories are calculated by the Computed Torque Control (CTC) method [23], the Jacobian, i.e., the matrix containing the muscular moment arms with respect to the joints, is evaluated, and the force limits for each muscle are set depending on the muscle model adopted (these limits are calculated only once per time step, right after the predictor). With all these elements, the force-sharing problem is addressed by optimization, as explained in Sect. 3, giving as result the muscular forces that allow building the vector of applied forces of the multibody system. It must be noted that, if the muscular forces are capable of fulfilling the torque constraints in the force-sharing problem, the resulting vector of applied forces of the multibody system will be identical to that previously calculated by the CTC method. Otherwise, one or more components of the vector among those corresponding to degrees of freedom driven by muscles will be different. This should not happen, as the movements were executed in practice and, hence, they were indeed feasible. However, the problem modeling could lead to such situations.

Finally, the linearized dynamic equations provided by the Newton–Raphson procedure are solved in the corrector block, the coordinates are updated, and a new iteration is carried out or not depending on the achieved convergence. Then, if a physiological model of the muscle has been adopted that includes a differential equation for contraction dynamics, a root solver is executed to find both the excitations that originated the obtained muscular forces and the activations at the new time step.

All the analyses were run on an Intel Core i7 6700K @ 4.00 GHz, 16 GB RAM, SSD 250 GB, with operating system Windows 10 Pro 20H2. The single-threaded program was written in C++ and compiled with MSVC 2019 in release configuration with the /O2 /Ob2 /MD flags. Eigen was used as the linear algebra package. To measure efficiency, the run-time was chosen, defined as the time required to run the program. Other programs were turned off during the executions. Each analysis was executed five times and the average value was obtained after checking that deviations were negligible.

As said in the introduction, two movements were studied, gait and vertical jump. Several subjects performed each movement several times. The study was reviewed by the in-

stitutional review board and the ethics approval was obtained. All subjects gave written informed consent for their participation. The movements were captured in a motion analysis lab (Fig. 2) equipped with 12 optical infrared cameras (Natural Point, OptiTrack FLEX 3, sampling at 100 Hz) that acquired the position of 37 reflective markers attached to the subject, two force plates (AMTI, AccuGait, sampling at 100 Hz), and nine surface EMG sensors that recorded the signals from nine muscles (tibialis anterior, vastus medialis, vastus lateralis, gastrocnemius medialis, gastrocnemius lateralis, semitendinosus, biceps femoris, gluteus maximus, and gluteus medius) of the right leg of the subject at 1 kHz (BTS, FREEEMG, Quincy, MA, USA).

In the case of gait, ten subjects (seven males, three females, age $42 \pm 16$ years, height $173 \pm 16$ cm, body mass $73 \pm 26$ kg) were recruited for the study. The subjects walked at their self-selected speed ($1.1 \pm 0.2$ m/s) along a walkway where the two force plates were embedded. In the case of vertical jump, seven subjects (three males, four females, age $34 \pm 11$ years, height $166 \pm 12$ cm, body mass $64 \pm 17$ kg) were recruited for the study. The subjects started from rest with one foot on each force plate, and were asked to jump vertically to clearly detach their feet from the ground, landing again with one foot on each force plate.

The captures were processed. First, an extended Kalman filter (EKF) was used to filter the marker trajectories and reconstruct the motion with a process noise variance of 1 m/s$^2$ and a cutoff frequency of 20 Hz [26]. Afterwards, inverse dynamics was applied by means of a velocity transformation method [27] implemented in the in-house developed MBSLIM library [28] programmed in FORTRAN, to get the joint torques as described in [29], and static optimization was carried out to obtain muscular forces [30]. The EMG signals were rectified, filtered by singular spectrum analysis (SSA) with a window length of 250 samples [31] (equivalent to the common forward and reverse low-pass 5th order Butterworth filter with a cut-off frequency of 6 Hz), and, then, normalized with respect to their maximal values. Then, the most consistent capture for each movement was selected for the study, based on a consistency index that was defined from contributions at kinematic level (minimum dispersion in geometric constraints satisfaction along the motion), at dynamic level (minimum residual in the base body), and at muscular level (best match with EMG signals [21]). More detail about this consistency index is provided in Appendix.

For gait, the selected capture was from a healthy adult female (Fig. 2(a)), age 30 years, height 165 cm, and body mass 50 kg, who walked at her self-selected speed of 0.92 m/s during 1.23 s. For vertical jump, the selected capture, of 1.90 s, was from a healthy adult female (Fig. 2(b)), age 29 years, height 157 cm, body mass 47 kg, who reached a height of 20 cm in her jump.

The multibody model (Fig. 3) was composed of 18 anatomical segments (two hindfeet, two forefeet, two shanks, two thighs, pelvis, torso, neck, head, two arms, two forearms, and two hands) connected by 17 spherical joints, leading to 57 degrees of freedom. Using spherical joints showed to be a good compromise between more simplified models for some joints (as revolute joints for the knees) and more detailed models for those joints (as surface–surface contacts for the knees), in the sense that it allowed a good fit of the captured kinematic data.

Six degrees of freedom were considered as actuated by muscles at each leg: the three rotations of the hip, the flexion/extension of the knee, and the dorsi/plantarflexion and the inversion/eversion of the ankle. Muscles were modeled as one or more straight-line segments with via points. These points corresponded to the attachments of muscle and tendon to bone and were defined as the origin (i.e., proximal attachment) and insertion (i.e., distal attachment). Muscle properties and local coordinates for these points were obtained

**Fig. 3** Multibody model: (**left**) skeletal level: bodies and joints; (**right**) musculoskeletal level, standard distribution composed of 43 muscles at each leg
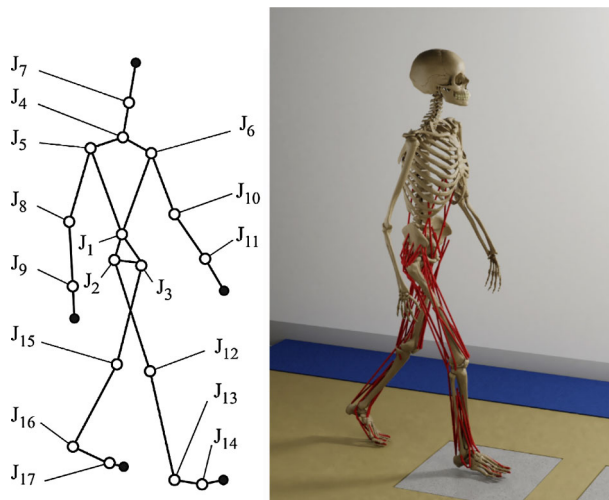
from OpenSim (model Gait2392) [32] and scaled to each subject from the generic reference OpenSim model, as commented in [30]. Two muscular distributions were defined: the standard one, composed of 43 muscles at each leg (listed in Table 2); and the simplified one, composed of eight muscles at each leg (glutei, hip flexors, hamstring, rectus femoris, vastii, gastrocnemius, soleus, tibialis anterior) [11].

## 3 Muscle models

This section is devoted to explain the way in which the force-sharing problem was addressed and to list and describe the four muscle models being compared.

### 3.1 Force-sharing problem

Looking again at Fig. 1, it can be seen that, once the joint torques required to track the captured joint trajectories, $\mathbf{Q}_{CTC}$, are calculated, it is necessary to solve the force-sharing problem for those degrees of freedom that are considered to be driven by muscles, i.e., those indicated in Sect. 2. The classical solution here is to pose an optimization problem with a cost function which aims to mimic the muscular recruitment strategy followed by the Central Nervous System (CNS), and with upper and lower bounds for the muscular forces. In this work, the optimization problem was set in the form:

$$
\begin{aligned}
\text{minimize} \quad & \sum_{i=1}^{m} \left( \frac{F_i}{F_{i,max}} \right)^2 \\
\text{subject to} \quad & \mathbf{J}^T \mathbf{F} = \mathbf{Q}_{CTC} \\
& F_{i,min} < F_i < F_{i,max}; \quad i = 1, \ldots, m
\end{aligned}
\tag{1}
$$

where $F_i$ stands for each muscle force, and $m$ is the number of modeled muscles. The limits for each muscle force, $F_{i,min}$ and $F_{i,max}$, depend on the muscular model selected, and will be provided in the next subsections.

**Fig. 4** The Hill muscle model. The muscle is formed by the contractile element (CE) and the passive parallel element (PE). The tendon is the serial element (SE)
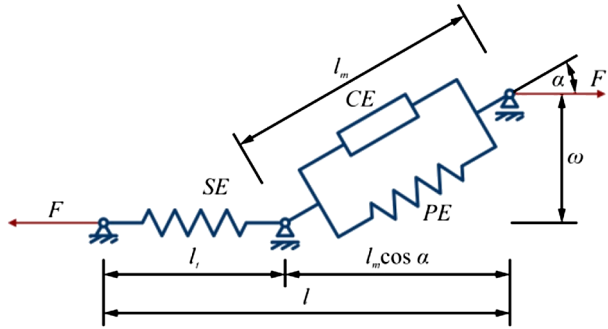


## 3.2 The full Hill muscle model

The Hill muscle model [33] assumes that the muscle fibers are straight, parallel, of equal length, coplanar, massless, frictionless, with fixed height, and attached to the tendon forming a pennation angle [20]. The Hill muscle model consists of three elements (Fig. 4): (i) a contractile element ($CE$) that models the active part of the muscle and generates a force that is a function of muscle activation, muscle length, and muscle velocity; (ii) a nonlinear passive element ($PE$) in parallel with the contractile element that models the passive behavior and elasticity of the muscle fiber; and (iii) a nonlinear passive element in series with the former two ($SE$) that models the tendon elasticity.

The force equilibrium equation is

$$F = F_{SE} = (F_{CE} + F_{PE}) \cos \alpha, \tag{2}$$

where $F$ is the musculotendon force, $F_{CE}$, $F_{PE}$, and $F_{SE}$ are the forces produced by the contractile, passive parallel, and passive serial element, respectively, and $\alpha$ is the pennation angle. These forces can be further detailed as

$$F_{CE} = a F_0 f_l(l_m) f_v(v_m); \quad F_{PE} = F_0 f_p(l_m); \quad F_{SE} = F_0 f_t(l_t), \tag{3}$$

where $a$ is the muscle activation, $F_0$ is the maximum isometric force, $f_l$ and $f_v$ are dimensionless force–length and force–velocity relationships, respectively, $f_p$ and $f_t$ are dimensionless force–length relationship of the passive parallel and passive serial element, respectively, $l_m$ and $l_t$ are the lengths of muscle and tendon, respectively, and $v_m$ is the velocity of muscle contraction.

Muscle activation, $a$, is a function of neural excitation, $u$, and can be modeled as a first order differential equation representing activation dynamics. Although different formulations can be found in the literature [18, 20], in this work the differential equation provided in [30] was substituted by its closed-form solution for the sake of efficiency:

$$a(t + \Delta t) = u(t) + [a(t) - u(t)] e^{-\frac{\Delta t}{\tau_a}}, \quad u > a$$
$$a(t + \Delta t) = u(t) + [a(t) - u(t)] e^{-\frac{\Delta t}{\tau_d}}, \quad u \leq a \tag{4}$$

where $\tau_a$ and $\tau_d$ are the activation and deactivation time constants, set to 15 and 50 ms, respectively.

The rate of change of the tendon force with respect to time is given by

$$\dot{F} = k_t v_t, \tag{5}$$

with $k_t$ being the variable tendon stiffness and $v_t$ the tendon elongation velocity.

A combination of (2), (3), and (5) leads to the first order differential equation representing contraction dynamics [34]

$$\dot{F}(t) = f(a(t), F(t), l(t), v(t)) = k_t \left[ v - \frac{v_{max}}{\cos \alpha} f_v^{-1} \left( \frac{\frac{F}{\cos \alpha} - F_0 f_p(l_m)}{a F_0 f_l(l_m)} \right) \right], \quad (6)$$

where $l$ and $v$ are the musculotendon length and velocity, respectively (Fig. 4), $v_{max}$ is the maximum muscle contraction velocity, and $f_v^{-1}$ is the inverse of function $f_v$. The details about all the terms involved in contraction dynamics can be found in [34].

To define the minimum and maximum physiologically feasible musculotendon force at a certain time point, (6) is integrated during the current time step for minimum ($u = 0$) and maximum ($u = 1$) muscle excitations, yielding $F_{min}$ and $F_{max}$ (see Fig. 1), respectively. The trapezoidal rule was used as integrator with a time-step size of 1 ms. For the initial time, it is assumed that muscle velocity is null, $v_m = 0$, so that $F_{min}$ and $F_{max}$ correspond to the minimum and maximum values of activation, $a = a_{min}$ and $a = 1$, respectively.

As said in Sect. 2 and illustrated in Fig. 1, if a physiological model of the muscle is adopted, as it is the case when using the described full Hill model, a root solver must be executed at each time step to find both the excitations that originated the obtained muscular forces and the activations at the new time step. The equations to be used within the root solver are (4), algebraic, and (6), differential. The iterative Newton's method was employed, their required derivatives being computed numerically by forward differences.

The integration of (6) presents some numerical problems. First, function $f_v^{-1}$ has numerical singularities. Second, the muscle can reach unrealistically short lengths. To avoid these problems, the same countermeasures described in [20] were implemented. The minimum value of activation was set to $a_{min} = 0.001$. In addition, a slight change was applied in the curvature of function $f_v^{-1}$ when $f_v$ approaches its maximum value, as described in [33], to streamline convergence. Furthermore, the use of an implicit integration scheme helps to overcome these problems [20].

### 3.3 The rigid-tendon model with activation dynamics

The rigid-tendon model adds to the Hill model the assumption that the tendon stiffness is so high that the tendon length variation can be neglected. This assumption greatly simplifies the model complexity, because it transforms contraction dynamics into an algebraic relationship which needs no integration, namely

$$F = [a F_0 f_l(l_m) f_v(v_m) + F_{PE}(l_m)] \cos \alpha. \quad (7)$$

In this case, the limits for the muscle force are directly obtained as

$$F_{min} = [a_{min} F_0 f_l(l_m) f_v(v_m) + F_{PE}(l_m)] \cos \alpha \\ F_{max} = [a_{max} F_0 f_l(l_m) f_v(v_m) + F_{PE}(l_m)] \cos \alpha \quad (8)$$

where the values of minimum and maximum activations, $a_{min}$ and $a_{max}$, are obtained from (4) for $u = 0$ and $u = 1$, respectively.

An additional consequence of the rigid-tendon assumption is that the execution of the root solver at the end of the time step (see Fig. 1) is not required.

### 3.4 The rigid-tendon model without activation dynamics

Another simplification explored in this work is not considering the delay between neural excitation and muscle activation, thus eliminating the need of using (4), since $a(t) = u(t)$.

In this case, the muscle force is also given by (7), and its limits are directly obtained as

$$
\begin{aligned}
F_{min} &= [F_{PE}(l_m)] \cos \alpha \\
F_{max} &= [F_0 f_l(l_m) f_v(v_m) + F_{PE}(l_m)] \cos \alpha.
\end{aligned}
\tag{9}
$$

And, of course, the execution of the root solver at the end of the time step is not required either.

### 3.5 A nonphysiological model

This option means not using a muscle model at all: the muscle forces in the force-sharing optimization are just limited to be in the range between zero and the maximum isometric force, but no further physiological limitations are imposed:

$$
\begin{aligned}
F_{min} &= 0, \\
F_{max} &= F_0.
\end{aligned}
\tag{10}
$$

Once again, there is no need of executing the root solver at the end of the time step.

## 4 Muscle torque generators

Muscle torque generators (MTGs) are functions intended to represent musculoskeletal torque at joint level. The goal is to reduce the model complexity and computational burden while keeping some level of biofidelity.

MTGs are typically adjusted for a particular case or movement, and different kinds of functions have been explored in the literature [9]. In this work, the functions proposed in [35] were used for the degrees of freedom driven by muscles, indicated at the end of Sect. 2. For each degree of freedom, the MTG has a different expression for each direction of torque exertion (agonist–antagonist). The maximum torque provided by the MTG is

$$
T_{max}\left(\theta, \dot{\theta}\right) =
\begin{cases}
C_1 \cos\left(C_2\left(\theta - C_3\right)\right) \left( \dfrac{2C_4 C_5 + \dot{\theta}\left(C_5 - 3C_4\right)}{2C_4 C_5 + \dot{\theta}\left(2C_5 - 4C_4\right)} \right), & \dot{\theta} \geq 0 \\[4mm]
C_1 \cos\left(C_2\left(\theta - C_3\right)\right) \left( \dfrac{2C_4 C_5 - \dot{\theta}\left(C_5 - 3C_4\right)}{2C_4 C_5 - \dot{\theta}\left(2C_5 - 4C_4\right)} \right) \left(1 - C_6 \dot{\theta}\right), & \dot{\theta} < 0
\end{cases}
\tag{11}
$$

where $\theta$ and $\dot{\theta}$ are the angle and angle velocity of the corresponding degree of freedom, and $C_i$, $i = 1, \ldots, 6$ are parameters that must be calibrated for each subject and degree of freedom. The torque provided by the MTG under a certain activation $a$, with $a$ ranging between 0 and 1, is

$$
T\left(\theta, \dot{\theta}\right) = a T_{max}\left(\theta, \dot{\theta}\right) + T_{pas}\left(\theta\right).
\tag{12}
$$

The passive torque, $T_{pas}$, due to muscular tissue, tendons, and ligaments was not considered, since passive torques act in the limits of motion ranges, which are not reached in the two movements studied.

To calibrate the parameters $C_i$, $i = 1, \ldots, 6$ of the MTGs that appear in (11), the rigid-tendon Hill model without activation dynamics was used for the muscles of the standard distribution listed in Table 2. For each degree of freedom, maximum activation was given to the muscles acting in one direction, while null activation was given to the muscles acting in the opposite direction, and the torque provided by the muscles under such conditions was calculated. This was repeated in the opposite motion direction of the degree of freedom. And the two calculations were repeated for the whole range of angular values of the degree of freedom and, for each angular value, for the whole range of feasible angular velocities. When calibrating the MTG for one degree of freedom, the remaining degrees of freedom were supposed to be fixed in their angular values, which correspond to the nominal standing position. A limitation of this approach is that it cannot take into account the combined effect of the position and velocity of several degrees of freedom at the same time, which leads to the inaccurate consideration of muscles that cross more than one joint.

Once the table containing the values of maximum torque for each degree of freedom as function of the angular position and velocity was built, the six parameters of each MTG were obtained by optimization, with the cost function being the root mean square (RMS) error and with the error calculated as the difference between the maximum torques provided by (11) and the torques from the table. Optimization was carried out in two steps: first, five global optimizations were run using the genetic algorithm *ga* from Matlab and, then, the solution with minimum value of the cost function was selected as initial guess for the gradient-based algorithm *fmincon* from Matlab.

It must be noted that, when using MTGs, there is no need of calculating some terms in the forward-dynamics algorithm of Fig. 1, as the Jacobian matrix containing the moment arms, the limits of muscular forces, and even the optimization devoted to solving the force-sharing problem. Instead, once the joint torques required to track the captured joint trajectories, $\mathbf{Q}_{CTC}$, are obtained, each element of vector $\mathbf{Q}_{CTC}$ corresponding to a degree of freedom driven by muscles is equaled to the MTG torque given by (12), so that the MTG activation can be worked out from that equation.

## 5 Muscular synergies

Muscular synergies reduce the problem dimensionality and, hence, they could be expected to increase the efficiency of the forward-dynamics algorithm depicted in Fig. 1. In this work, the synergies were extracted, for each movement studied, from the activations obtained by static optimization from the corresponding motion capture with the standard muscle distribution, as done in [30]. Calling $n_s$ the number of synergies adopted, the activations can be expressed as

$$\mathbf{a}(t) = \mathbf{W}\mathbf{s}(t), \tag{13}$$

where $\mathbf{a}(m \times 1)$ is the vector of muscular activations, $\mathbf{W}(m \times n_s)$ is the matrix of synergy vectors (or weights), and $\mathbf{s}(n_s \times 1)$ is the vector of synergy controls. Each synergy control is a B-spline defined by $p = (f - 1)/5 + 1$ nodal points, with $f$ being the number of frames (time steps) of the motion capture. Therefore, to obtain the synergy controls and the synergy vectors, an optimization is carried out with $n_s \times (p + m)$ design variables, a set of linear equality constraints to enforce that the sum of weights for each synergy is 1, lower bounds to impose that synergy controls and vectors are nonnegative, and a cost function that takes into account: (i) the deviation between the joint torques provided by muscles and from inverse dynamics; (ii) the values of the activations; and (iii) a penalization term to ensure that

activations stay between 0 and 1. Optimization was carried out in two steps: first, five global optimizations were run using the genetic algorithm *ga* from Matlab and, then, the solution with minimum value of the cost function was selected as initial guess for the gradient-based algorithm *fmincon* from Matlab. The problem was solved for 2 to 6 synergies.

The changes required to adapt the forward-dynamics algorithm of Fig. 1 to the use of synergies are described in what follows, pointing out first that the rigid-tendon model without activation dynamics was considered here, since the synergies had been obtained from a static optimization.

As seen in Sect. 3, the muscle force can be expressed in terms of the activation as

$$F = (F_{CE} + F_{PE}) \cos \alpha = a F_0 f_l(l) f_v(v) \cos \alpha + F_{PE}(l) \cos \alpha. \tag{14}$$

Grouping terms and moving to vectorial form gives

$$\mathbf{F} = \mathbf{D}\mathbf{a} + \mathbf{F}_{PE}^*, \tag{15}$$

where $\mathbf{D} = diag\,(a_i F_{0i} f_l(l_i) f_v(v_i) \cos \alpha_i)$, $i = 1, \ldots, m$ is a diagonal matrix and the terms of the column vector $\mathbf{F}_{PE}^*$ have the form $F_{PEi}^* = F_{PEi} \cos \alpha_i$, $i = 1, \ldots, m$. Introducing now (13) into (15) leads to

$$\mathbf{F} = \mathbf{D}\mathbf{W}\mathbf{s} + \mathbf{F}_{PE}^*, \tag{16}$$

which provides the muscular forces as functions of the synergy controls.

In the optimization for the force-sharing problem, the cost function given in (1) can be written in matrix form as

$$C = \sum_{i=1}^{m} \left( \frac{F_i}{F_{i,max}} \right)^2 = \frac{1}{2} \mathbf{F}^T \mathbf{H} \mathbf{F}, \quad \text{with} \quad \mathbf{H} = diag \left( \frac{2}{F_{i,max}^2} \right), \quad i = 1, \ldots, m. \tag{17}$$

Substitution of (16) in (17) yields

$$C = \frac{1}{2} \mathbf{F}^T \mathbf{H} \mathbf{F} = \frac{1}{2} \mathbf{s}^T \mathbf{E} \mathbf{s} + b, \tag{18}$$

with $\mathbf{E} = \mathbf{W}^T \mathbf{D}^T \mathbf{H} \mathbf{D} \mathbf{W}$ and $b = \frac{1}{2} \mathbf{F}_{PE}^{*T} \mathbf{H} \mathbf{F}_{PE}^*$. Since the optimization for the force-sharing problem is conducted, at each iteration within a certain time step, for given positions and velocities of the system, the value of $b$ is constant and, hence, it can be eliminated from the cost function.

The constraints of the joint torques in the degrees of freedom driven by muscles imposed in the optimization for the force-sharing problem (1) can also be transformed by introducing (16), namely

$$\mathbf{J}^T \mathbf{F} = \mathbf{Q}_{CTC} \quad \Rightarrow \quad \mathbf{J}^T \left( \mathbf{D}\mathbf{W}\mathbf{s} + \mathbf{F}_{PE}^* \right) = \mathbf{Q}_{CTC} \quad \Rightarrow \quad \mathbf{G}^T \mathbf{s} = \mathbf{Q}_{CTC}^*, \tag{19}$$

where $\mathbf{G}^T = \mathbf{J}^T \mathbf{D} \mathbf{W}$ and $\mathbf{Q}_{CTC}^* = \mathbf{Q}_{CTC} - \mathbf{J}^T \mathbf{F}_{PE}^*$.

Therefore, the optimization for the force-sharing problem shown in (1) can be stated, having now as design variables the synergy controls, as follows:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2} \mathbf{s}^T \mathbf{E} \mathbf{s} \\ \text{subject to} \quad & \mathbf{G}^T \mathbf{s} = \mathbf{Q}_{CTC}^* \\ & 0 < s_i < 1; \quad i = 1, \ldots, n_s \end{aligned} \tag{20}$$

# 6 Interpolation of configuration-dependent muscular magnitudes

Solving muscle dynamics involves a high computational cost. A way to reduce this cost is by precomputing some kinematic magnitudes that are configuration-dependent and tabulate them. For example, muscular moment arms require an expensive calculation and are configuration-dependent. Other magnitudes suitable for precomputation are muscular lengths, which could also represent a time consuming calculation if complex wrapping models were used [36]. Evaluation of muscular velocities can also be speeded up by tabulating the derivatives of muscular lengths with respect to joint degrees of freedom, which are also needed for the calculation of moment arms.

To generate the tables, the degrees of freedom driven by muscles were given a set of values within their ranges of motion, and the mentioned magnitudes were precomputed. The major drawback of this approach is the dimensionality problem, as the size of the table grows exponentially with the number of degrees of freedom affecting the magnitude, and this becomes especially critical for muscles that cross more than one joint.

In this work, moment arms, lengths and length derivatives with respect to degrees of freedom were tabulated for the 43 muscles of each leg conforming the standard distribution, with several table resolutions. Linear interpolation was applied.

# 7 Results and discussion

Once all the alternatives have been described, the obtained results are to be shown and discussed for the two exercises considered, i.e., gait and vertical jump.

## 7.1 Muscle models

The standard distribution of 43 muscles in the right leg is selected to show the accuracy offered by the four muscle models considered in this study. The full Hill model (H1) is taken as reference, and the discrepancies with respect to the reference of the muscular excitations, activations and normalized forces obtained when using the rigid-tendon model with activation dynamics (H2), the rigid-tendon model without activation dynamics (H3) and the nonphysiological model (NP) provide the corresponding errors. The error in each magnitude is calculated as the root mean square (RMS) of the difference between its value with the corresponding muscle model and the value with the full Hill model for all the time steps of the exercise, $n_t$,

$$e_i^u = \sqrt{\frac{\sum_{j=1}^{n_t}\left(u_{ij} - u_{ij}^{ref}\right)^2}{n_t}}; \quad e_i^a = \sqrt{\frac{\sum_{j=1}^{n_t}\left(a_{ij} - a_{ij}^{ref}\right)^2}{n_t}}; \quad e_i^F = \sqrt{\frac{\sum_{j=1}^{n_t}\left(\frac{F_{ij} - F_{ij}^{ref}}{F_{max,i}}\right)^2}{n_t}}$$

$$i = 1, \ldots, m, \tag{21}$$

where $e_i^u$, $e_i^a$, and $e_i^F$ are the RMS errors (RMSE) for the excitation, activation, and normalized force, respectively, of muscle $i$, with $m$ being the number of muscles, $m = 43$ for the standard distribution in the right leg, and the *ref* superscript indicates the values for the full Hill model, taken as reference. The value $F_{max,i}$, used to normalize the force of each muscle, is the maximum force value generated by the corresponding muscle (full Hill model) along the motion considered.

**Table 1** Comparison of accuracy in aggregated muscular excitations, activations, and normalized forces for the four muscle models considered: the full Hill model (H1) taken as reference, the rigid-tendon model with activation dynamics (H2), the rigid-tendon model without activation dynamics (H3), and a nonphysiological model (NP). The distinction between long-tendon and short-tendon muscles is also provided

| | GAIT | | | VERTICAL JUMP | | |
|---|---|---|---|---|---|---|
| | H2 | H3 | NP | H2 | H3 | NP |
| $e^u$ | 0.069 | 0.089 | – | 0.049 | 0.076 | – |
| $e^a$ | 0.051 | 0.070 | – | 0.031 | 0.061 | – |
| $e^F$ | 0.269 | 0.341 | 0.333 | 0.189 | 0.240 | 0.223 |
| $e^F$ *(long)* | 0.328 | 0.416 | 0.404 | 0.238 | 0.282 | 0.255 |
| $e^F$ *(short)* | 0.147 | 0.186 | 0.187 | 0.089 | 0.152 | 0.156 |

Table 1 shows the aggregated RMSE of excitations, activations and normalized forces for the 43 muscles of the standard distribution in the right leg, obtained as

$$e^u = \frac{\sum_{i=1}^{43} e_i^u}{43}; \qquad e^a = \frac{\sum_{i=1}^{43} e_i^a}{43}; \qquad e^F = \frac{\sum_{i=1}^{43} e_i^F}{43} \qquad (22)$$

for the two movements considered in the study, namely gait and vertical jump. Note that excitations and activations are magnitudes between 0 and 1. The table has no entries for the RMSE of excitations and activations in the case of the nonphysiological model (NP) of muscle as only forces are considered in this case. Additionally, the table shows separately the aggregated RMSE of normalized muscular forces due to muscles with long tendon and short tendon. Long- and short-tendon muscles are here defined as those whose tendon slack length is longer and shorter, respectively, than the optimum muscle fiber length [20].

Table 2 details the discrepancies among the four models for the 43 muscles, but focusing only on the normalized muscular forces, for the sake of simplicity. Long-tendon muscles have been written in italics to distinguish them from short-tendon muscles. And Table 3 gathers the run-times required to run the forward-dynamics analysis for each muscular modeling alternative.

It can be seen that the rigid-tendon model with activation dynamics (H2) shows a good balance between efficiency (Table 3) and accuracy (Table 1). Models H3 and NP provide marginal improvements in efficiency with significant reductions in accuracy. Going into more detail, it can be observed that the rigid-tendon model is more suitable for short-tendon muscles, as already pointed out in [20]. In fact, there are some muscles which show very high errors in Table 2, all of them having very high ratios of tendon length with respect to muscle length. Therefore, it could even be considered the option of applying the full Hill model to long-tendon muscles and the rigid-tendon model to short-tendon muscles. There is no problem in combining the two modeling options in the same simulation. It can also be observed that the rigid-tendon model without activation dynamics (H3) is not superior to the nonphysiological model (NP). Therefore, using the model H3 is not useful.

To provide a better interpretation of the values in the previous tables, Fig. 5 shows the histories of muscle forces obtained with the four models in the case of two muscles for each movement considered in the study. The right rectus femoris is a case of long-tendon muscle. The errors in normalized force provided by Table 2 for this muscle are 0.096 (H2), 0.146 (H3), and 0.141 (NP) for gait, and 0.030 (H2), 0.142 (H3), and 0.140 (NP) for vertical jump. No significant differences are found in the errors of H3 and NP for the two activities. However, the error of H2 is much lower for the more dynamic motion of vertical jump. The error.

**Table 2** Comparison of accuracy in muscular forces for the four muscle models considered: the full Hill model (H1), taken as reference, the rigid-tendon model with activation dynamics (H2), the rigid-tendon model without activation dynamics (H3), and a nonphysiological model (NP). The entries of the table are the RMSE of normalized muscle forces, denoted before as $e_i^F$, $i = 1, \ldots, 43$. Long-tendon muscles are in italics. Error values that are larger than twice the average error value for the corresponding muscle model (H2, H3, NP) and muscle type (long- or short-tendon) have been highlighted in bold

| Muscles | GAIT | | | VERTICAL JUMP | | |
|---|---|---|---|---|---|---|
| | H2 | H3 | NP | H2 | H3 | NP |
| *R. Gluteus Medius Anterior* | 0.059 | 0.138 | 0.124 | 0.062 | 0.154 | 0.148 |
| R. Gluteus Medius Middle | 0.095 | 0.159 | 0.166 | 0.056 | 0.119 | 0.125 |
| R. Gluteus Medius Posterior | 0.132 | 0.205 | 0.171 | 0.066 | 0.125 | 0.133 |
| R. Gluteus Minimus Anterior | 0.197 | 0.143 | 0.177 | 0.123 | 0.149 | 0.155 |
| R. Gluteus Minimus Middle | 0.133 | 0.191 | 0.129 | 0.068 | 0.095 | 0.099 |
| R. Gluteus Minimus Posterior | 0.118 | 0.228 | 0.186 | 0.070 | 0.094 | 0.096 |
| *R. Gluteus Max. Ant. (sup.)* | 0.211 | 0.368 | 0.430 | 0.041 | 0.076 | 0.095 |
| *R. Gluteus Maximus Middle* | 0.094 | 0.105 | 0.105 | 0.033 | 0.089 | 0.094 |
| *R. Gluteus Max. Post. (inf.)* | 0.086 | 0.182 | 0.177 | 0.084 | 0.175 | 0.224 |
| *R. Adductor Longus* | 0.038 | 0.137 | 0.137 | 0.078 | 0.205 | 0.205 |
| R. Adductor Brevis | 0.119 | 0.170 | 0.175 | 0.118 | 0.175 | 0.172 |
| R. Adductor Magnus Superior | 0.116 | 0.167 | 0.177 | 0.079 | 0.210 | 0.212 |
| *R. Adductor Magnus Middle* | 0.079 | 0.183 | 0.182 | 0.079 | 0.205 | 0.218 |
| *R. Adductor Magnus Inferior* | 0.139 | 0.179 | 0.177 | 0.358 | 0.502 | 0.470 |
| R. Tensor Fasciae Latae | **2.347** | **2.396** | **1.315** | **0.492** | 0.410 | 0.096 |
| R. Pectineus | 0.091 | 0.129 | 0.129 | 0.141 | 0.205 | 0.205 |
| R. Iliacus | 0.252 | 0.260 | 0.218 | 0.072 | 0.175 | 0.178 |
| *R. Psoas* | 0.232 | 0.270 | 0.249 | 0.109 | 0.227 | 0.228 |
| *R. Quadratus femoris* | 0.089 | 0.176 | 0.166 | 0.060 | 0.143 | 0.144 |
| *R. Gemellus* | 0.145 | 0.172 | 0.167 | 0.096 | 0.144 | 0.145 |
| *R. Piriformis* | **2.872** | **3.974** | **4.522** | 0.056 | 0.088 | 0.132 |
| R. Semitendinosus | 0.047 | 0.165 | 0.215 | 0.058 | 0.189 | 0.199 |
| R. Semimembranosus | 0.109 | 0.205 | 0.204 | 0.075 | 0.189 | 0.195 |
| *R. Biceps Femoris Long Head* | 0.095 | 0.226 | 0.240 | **1.053** | **1.011** | **0.939** |
| R. Biceps Femoris Short Head | 0.173 | 0.218 | 0.250 | 0.079 | 0.177 | 0.176 |
| R. Sartorius | 0.277 | 0.228 | 0.283 | 0.113 | 0.132 | 0.155 |
| R. Gracilis | 0.034 | 0.102 | 0.166 | 0.106 | 0.202 | 0.204 |
| *R. Rectus Femoris* | 0.096 | 0.146 | 0.141 | 0.030 | 0.142 | 0.140 |
| *R. Vastus Medialis* | 0.042 | 0.105 | 0.097 | 0.020 | 0.038 | 0.055 |
| *R. Vastus Intermedius* | 0.049 | 0.106 | 0.097 | 0.035 | 0.052 | 0.068 |
| *R. Vastus Lateralis* | 0.044 | 0.060 | 0.068 | 0.026 | 0.044 | 0.090 |
| *R. Gastrocnemius Medial* | **0.890** | 0.784 | 0.731 | 0.081 | 0.162 | 0.161 |
| *R. Gastrocnemius Lateral* | 0.098 | 0.108 | 0.122 | 0.060 | 0.164 | 0.152 |
| *R. Soleus* | 0.228 | 0.192 | 0.173 | 0.008 | 0.071 | 0.080 |
| *R. Tibialis Posterior* | 0.067 | 0.086 | 0.085 | 0.034 | 0.146 | 0.145 |
| *R. Tibialis Anterior* | 0.107 | 0.186 | 0.197 | 0.023 | 0.169 | 0.166 |
| *R. Flexor Digitorum Longus* | 0.157 | 0.145 | 0.284 | **2.398** | **1.742** | **1.358** |
| *R. Flexor Hallucis Longus* | 0.619 | 0.780 | 0.691 | **1.325** | **1.116** | **0.926** |
| *R. Extensor Digit. Longus* | 0.120 | 0.239 | 0.268 | 0.027 | 0.194 | 0.192 |
| *R. Extensor Hallucis Longus* | 0.138 | 0.182 | 0.207 | 0.085 | 0.127 | 0.128 |
| *R. Peroneus Brevis* | 0.370 | 0.164 | 0.186 | 0.042 | 0.179 | 0.178 |
| *R. Peroneus Longus* | 0.052 | 0.173 | 0.188 | 0.031 | 0.173 | 0.171 |
| R. Peroneus Tertius | 0.122 | 0.126 | 0.158 | 0.098 | 0.134 | 0.134 |

**Table 3** Comparison of efficiency for the four muscle models considered: the full Hill model (H1), the rigid-tendon model with activation dynamics (H2), the rigid-tendon model without activation dynamics (H3), and a nonphysiological model (NP). The run-times are given in s. The duration of each movement is indicated in the first row

| GAIT (1.23 s) | | | | VERTICAL JUMP (1.90 s) | | | |
|---|---|---|---|---|---|---|---|
| H1 | H2 | H3 | NP | H1 | H2 | H3 | NP |
| 0.453 | 0.120 | 0.113 | 0.108 | 0.750 | 0.239 | 0.218 | 0.217 |
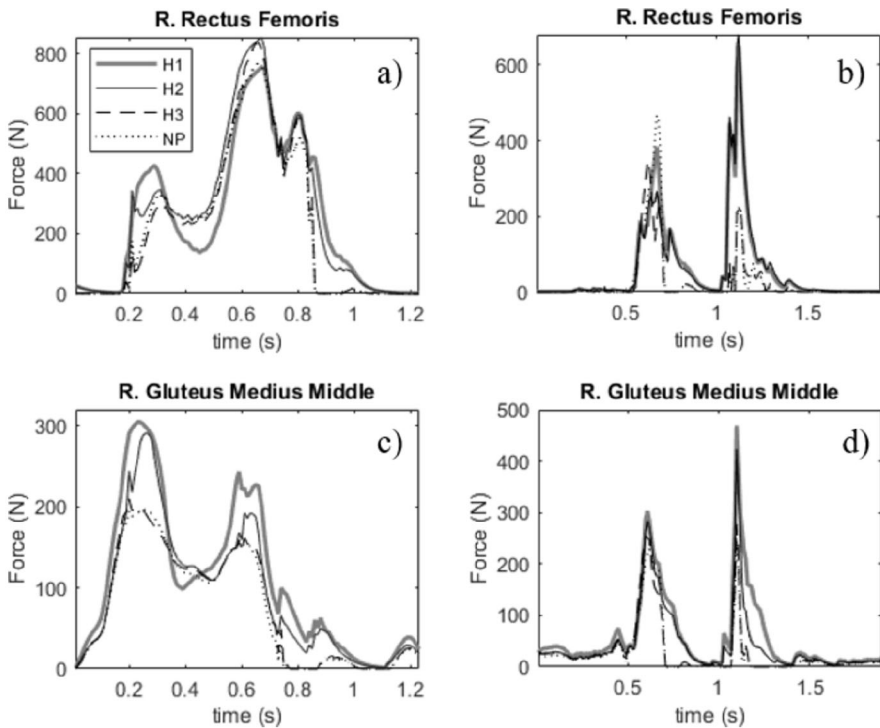


**Fig. 5** Histories of muscle forces with the four muscle models: (**a**) Right rectus femoris in gait; (**b**) Right rectus femoris in vertical jump; (**c**) Right gluteus medius middle in gait; (**d**) Right gluteus medius middle in vertical jump. Right rectus femoris is a long-tendon muscle while right gluteus medius middle is a short-tendon muscle

The right gluteus medius middle is a case of short-tendon muscle. The errors in normalized force provided by Table 2 for this muscle are 0.095 (H2), 0.159 (H3), and 0.166 (NP) for gait, and 0.056 (H2), 0.119 (H3), and 0.125 (NP) for vertical jump. This time, the errors for the three models (H2, H3, NP) are lower for the more dynamic motion of vertical jump. This fact is also confirmed by the results gathered in Table 1. Although the way of calculating the errors can favor vertical jump, as it presents muscular activity during only a part of the total duration of the experiment, the plots in Fig. 5 show a good behavior of the models for this movement. This may indicate that it is easier to model the muscular activity of a simpler (more two-dimensional) exercise, as the vertical jump, than a more complex (more three-dimensional) one, as walking, even if the latter is less dynamic.

**Table 4** Comparison of efficiency with respect to the number of muscles modeled for the four muscle models considered. The run-times given in the table are in s

| # of muscles | GAIT (1.23 s) | | | | VERTICAL JUMP (1.90 s) | | | |
|---|---|---|---|---|---|---|---|---|
| | H1 | H2 | H3 | NP | H1 | H2 | H3 | NP |
| 86 | 1.077 | 0.256 | 0.240 | 0.237 | 1.831 | 0.518 | 0.471 | 0.451 |
| 43 | 0.453 | 0.120 | 0.113 | 0.108 | 0.750 | 0.239 | 0.218 | 0.217 |
| 16 | 0.151 | 0.040 | 0.038 | 0.036 | 0.253 | 0.080 | 0.075 | 0.070 |
| 8 | 0.072 | 0.026 | 0.025 | 0.023 | 0.131 | 0.053 | 0.050 | 0.047 |
| 0 | 0.01 | | | | 0.02 | | | |
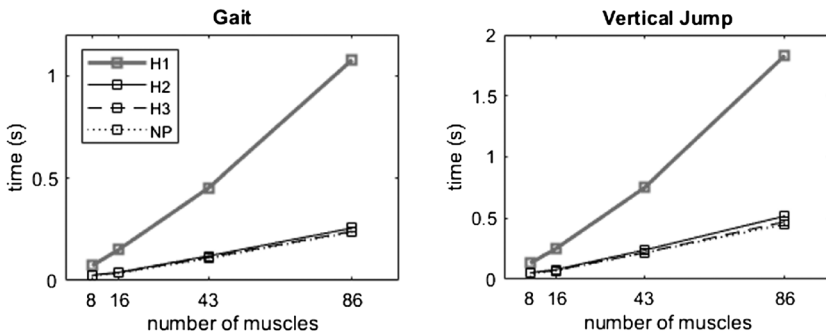


**Fig. 6** Relation between number of muscles and run-time needed to execute the forward-dynamics algorithm, for the four muscle models adopted

## 7.2 Number of muscles

Table 4 shows the influence of the number of muscles considered in the model on the efficiency of the simulation for the four muscle models considered. Four cases were compared: (i) the standard model of 43 muscles in both legs, i.e., a model with 86 muscles; (ii) the standard model in the right leg only, i.e., a model with 43 muscles; (iii) the simplified model in both legs, i.e., a model with 16 muscles; and (iv) the simplified model in the right leg only, i.e., a model with 8 muscles. The run-times required for the simulation in the skeletal case (no muscles) were 0.01 s for gait and 0.02 s for vertical jump.

It can be seen that including muscles increases the required run-times. However, such increase can range from two to four times with respect to the skeletal case for a moderate number of muscles (8 or 16) and the simplified muscle models (H2, H3, NP), to two orders of magnitude for a large number of muscles (43 or 86) and the full Hill model (H1). Once muscles are included in the model, the required run-time scales more than linearly with the number of muscles.

Figure 6 illustrates the obtained relation between number of muscles and run-time required to carry out the simulation of each movement considered in the study, gait and vertical jump, and for the four muscle models adopted. It can be observed that, as said before, all the models scale more than linearly with the number of muscles, and that the full Hill model shows a notably higher growth in run-time as the number of muscles increase than its counterparts, which in turn offer almost identical trends.

It must be noted that, in the case of 8 muscles per leg, only three degrees of freedom are driven by muscles in each leg (the angles in the sagittal plane of hip, knee, and ankle), instead

**Table 5** Comparison of accuracy as the RMSE between the activations provided by each MTG and the weighted average activations provided by the muscles contributing to it when using the rigid-tendon model without activation dynamics (H3) and the standard distribution of 43 muscles in the right leg

| MTG # | Agonist/Antagonist | GAIT | V. JUMP |
|---|---|---|---|
| 1 | Hip adduction | 0.0111 | 0.0050 |
| | Hip abduction | 0.0581 | 0.0773 |
| 2 | Hip flexion | 0.0247 | 0.0279 |
| | Hip extension | 0.0718 | 0.0242 |
| 3 | Hip internal rotation | 0.2126 | 0.0259 |
| | Hip external rotation | 0.1143 | 0.0610 |
| 4 | Knee flexion | 0.0137 | 0.0144 |
| | Knee extension | 0.0477 | 0.0783 |
| 5 | Ankle inversion | 0.0101 | 0.0234 |
| | Ankle eversion | 0.0068 | 0.0312 |
| 6 | Ankle plantarflexion | 0.0270 | 0.0429 |
| | Ankle dorsiflexion | 0.0216 | 0.0034 |
| Mean | – | 0.0516 | 0.0346 |

of the six degrees of freedom driven by muscles in each leg when modeling 43 muscles per leg. More degrees of freedom driven by muscles means more work for the forward-dynamics algorithm, apart from that strictly due to the increase in the number of muscles, as a higher number of torque constraints in the optimization for the force-sharing problem, or a higher number of columns in the Jacobian containing the moment arms.

### 7.3 Muscle torque generators

Six MTGs (each one with the agonist and antagonist components) were used for the six degrees of freedom driven by muscles in the right leg. It was found that the efficiency provided by the use of MTGs was almost the same achieved in the skeletal case: the run-times required to run the simulations were 0.01 s for gait and 0.02 s for vertical jump, exactly the same as in the skeletal case.

To give an idea of the accuracy provided by the use of MTGs, the activation of each MTG was compared with the weighted average activation of the muscles that contribute to that MTG when using the rigid-tendon model without activation dynamics in the standard model of 43 muscles for the right leg (the modeling employed to calibrate the MTGs). The weighted average activation of the muscles that contribute to an MTG was obtained as follows:

$$\bar{a} = \frac{\sum_{i=1}^{m} a_i F_{0i} r_i}{\sum_{i=1}^{m} F_{0i} r_i}, \tag{23}$$

where $a_i$ are the activations of the $m$ muscles contributing to that MTG, $F_{0i}$ are the maximum isometric forces of the muscles, and $r_i$ are the moment arms of the muscles.

Table 5 gathers the RMSE obtained from the difference, for each MTG, between both magnitudes, $a$ and $\bar{a}$, along the duration of the movements, while Fig. 7 details such discrepancies for two MTGs, the hip adductor and the knee extensor. It can be seen in Table 5 that the mean error is around 5% in gait and around 3.5% in vertical jump. Once again, it seems that the simplicity of vertical jump prevails over its high dynamics, leading to a better behavior of the simplified models than in the case of gait.
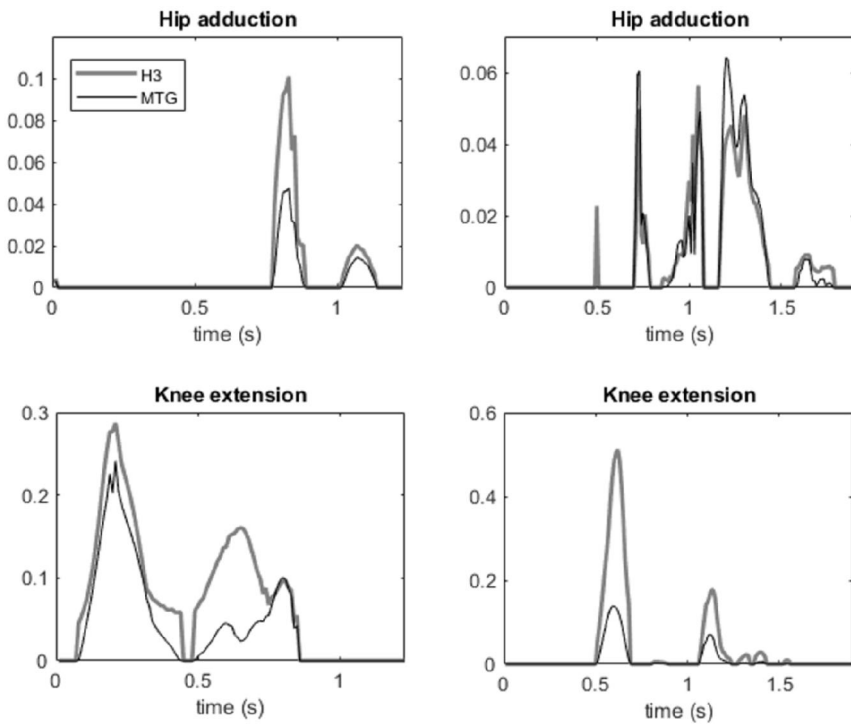
**Fig. 7** Histories of activations of two MTGs vs. weighted average activations of the muscles contributing to them, for gait (left) and vertical jump (right)

### 7.4 Muscular synergies

It was seen in Sect. 5 that muscular activations and forces can be recovered from the synergy controls by means of Eqs. (13) and (16). It was also said that the forward-dynamics analysis of gait and vertical jump was solved with a number of synergies ranging from 2 to 6.

The results obtained when using synergies were compared with the results obtained with the rigid-tendon model without activation dynamics (H3) for the standard distribution of 43 muscles in the right leg (the muscle model used to state the force-sharing problem in terms of synergies in Sect. 5).

Table 6 gathers the mean RMSE of aggregated muscular activations and normalized forces (normalized by dividing the force by the maximum isometric force) when using 2 to 6 synergies. The RMSE for each muscle was obtained from the difference between the history of the corresponding magnitude in the case of using synergies and the history of the same magnitude when using muscles modeled by method H3. Then, the mean RMSE was calculated as the mean of the RMSE of the 43 muscles. Figure 8 details the differences in force for two particular muscles.

When using muscles, the muscular forces are capable of yielding the torques, in the degrees of freedom driven by muscles, that are required by the CTC controller to track the captured movements. However, this may not occur, as pointed out in Sect. 2, for some modeling options. And this is the case when using synergies. Table 7 presents the RMSE between the torques required by the CTC method and the torques provided by the muscles when using the synergy-based approach, for the number of synergies ranging between 2 and

**Table 6** Comparison of accuracy as the RMSE between the aggregated activations and normalized forces obtained when using synergies and the same magnitudes obtained when using muscles modeled by method H3

| # of synergies | GAIT | | VERTICAL JUMP | |
|---|---|---|---|---|
| | Activation | Norm. Force | Activation | Norm. Force |
| 2 | 0.0886 | 0.4907 | 0.0848 | 0.6576 |
| 3 | 0.0919 | 0.5637 | 0.0762 | 0.4847 |
| 4 | 0.0819 | 0.9299 | 0.0645 | 0.4277 |
| 5 | 0.0832 | 1.2619 | 0.0695 | 0.4588 |
| 6 | 0.0663 | 0.8146 | 0.0816 | 0.5553 |
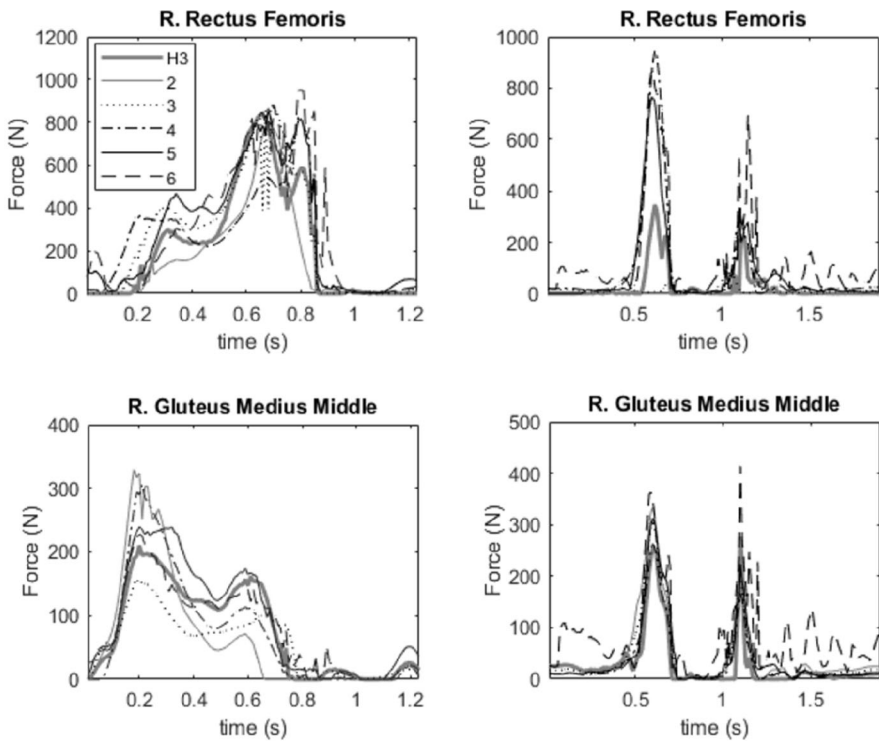


**Fig. 8** Histories of muscle forces obtained with the synergy-based approach for a number of synergies ranging from 2 to 6 compared to those obtained with the muscle-based approach and the muscles modeled by method H3. The plots on the left correspond to gait and the plots on the right correspond to vertical jump

6. In each case, the presented RMSE is the mean of the RMSE of the six degrees of freedom driven by muscles. Figure 9 details these discrepancies.

It can be seen in Table 7 and Fig. 9 that the satisfaction of the torque constraints in the force-sharing problem is better as more synergies are considered. This happens because the optimizer has more freedom to fulfill the constraints as the number of synergies increases. The problem is that the optimizer gives priority to the satisfaction of the constraints, putting less effort in minimizing the cost function. This is why the accuracy of muscular activations

**Table 7** RMSE in the satisfaction of torque constraints in the force-sharing problem when using synergies. The presented values are RMSE expressed in Nm

| # of synergies | GAIT | V. JUMP |
|---|---|---|
| 2 | 5.38 | 4.30 |
| 3 | 4.29 | 3.59 |
| 4 | 2.88 | 2.54 |
| 5 | 1.68 | 1.86 |
| 6 | 1.60 | 1.02 |

**Table 8** Efficiency achieved when using synergies. The run-times are given in s

| # of synergies | GAIT | V. JUMP |
|---|---|---|
| 2 | 0.0907 | 0.1758 |
| 3 | 0.0914 | 0.1767 |
| 4 | 0.0917 | 0.1779 |
| 5 | 0.0918 | 0.1817 |
| 6 | 0.0981 | 0.2016 |
| No synergies | 0.113 | 0.218 |

**Table 9** Comparison of accuracy in aggregated muscular excitations, activations and normalized forces when using tables with two resolutions for the calculation of configuration-dependent magnitudes

| Resolution | GAIT | | VERTICAL JUMP | |
|---|---|---|---|---|
| | 5° | 10° | 5° | 10° |
| $e^u$ ($\times 10^{-3}$) | 1.22 | 2.13 | 6.65 | 10.77 |
| $e^a$ ($\times 10^{-3}$) | 0.54 | 1.26 | 2.94 | 4.95 |
| $e^F$ ($\times 10^{-3}$) | 4.55 | 6.21 | 7.95 | 15.70 |

and forces does not improve with the number of synergies considered (see Table 6), as it happens with the level of constraint satisfaction.

Table 8 shows the run-times required to carry out the forward-dynamics analysis for the two movements considered in the study.

The run-times provided in Table 8 are similar to those shown in Table 4 for the model with 43 muscles in the right leg and using the rigid-tendon model without activation dynamics (H3). For gait, the run-time was 0.113 s, and, for vertical jump, the run-time was 0.218 s. Therefore, for this problem there is little advantage in using synergies from the efficiency point of view (a maximum gain of 20% for both movements), and the results are less accurate, as previously shown.

### 7.5 Interpolation of configuration-dependent muscular magnitudes

As said in Sect. 6, moment arms, lengths and length derivatives with respect to degrees of freedom were tabulated for the 43 muscles of each leg conforming the standard distribution, with several table resolutions. Linear interpolation was applied.

Table 9 gathers the RMSE in aggregated excitations, activations and normalized forces, obtained with expressions (21) and (22), when using tables with two resolutions (5° and 10°) and having as reference the values obtained without tables. In all cases, the full Hill model (H1) was employed as muscle model.

It can be observed in Table 9 that errors are very low, being higher in vertical jump than in gait. It can also be seen that reducing the resolution leads to higher errors, with an

**Table 10** Effect of the resolution in the size of the tables (in MB) used for the calculation of configuration-dependent magnitudes

| Resolution | GAIT | V. JUMP |
|---|---|---|
| 5° | 32.50 | 81.00 |
| 10° | 1.49 | 1.77 |

**Table 11** Effect of the resolution in the efficiency of the forward-dynamics analysis. The run-times are given in s

| Resolution | GAIT (1.23 s) | | | | VERTICAL JUMP (1.90 s) | | | |
|---|---|---|---|---|---|---|---|---|
| | H1 | H2 | H3 | NP | H1 | H2 | H3 | NP |
| 5° | 0.346 | 0.049 | 0.045 | 0.040 | 0.627 | 0.102 | 0.087 | 0.080 |
| 10° | 0.344 | 0.049 | 0.045 | 0.040 | 0.627 | 0.098 | 0.087 | 0.080 |
| No tables | 0.453 | 0.120 | 0.113 | 0.108 | 0.750 | 0.239 | 0.218 | 0.217 |
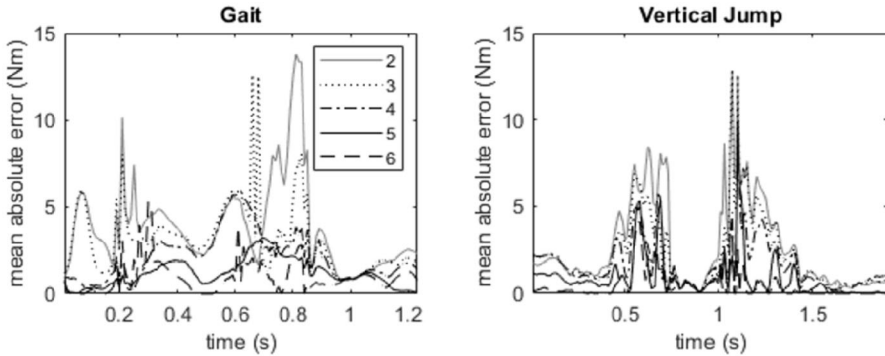


**Fig. 9** Histories of mean absolute errors in the satisfaction of the torque constraints in the force-sharing problem when using synergies

approximately linear trend in the analyzed range of resolution (since a null error could be expected for a resolution of 0°). However, this reduction in resolution entails an exponential reduction in the size of the tables, as shown in Table 10.

Finally, Table 11 gathers the run-times employed to carry out the forward-dynamics analysis of the two movements considered when using tables, along with the run-times required when not using them.

As it can be observed in Table 11, non-negligible advantages are obtained from the use of tables, being almost independent on the adopted resolution. The lower advantage obtained for H1 is reflecting the lower weight of the calculation of moment arms, lengths and length derivatives with respect to degrees of freedom in the whole forward-dynamics algorithm (see Fig. 1): the computation of the muscle force limits, the optimization for the force-sharing problem and the root-solver are much more computationally demanding. However, for H2, H3 and NP, these blocks are either less burdensome or even unnecesary. Gains in efficiency are between 16% and 24% for H1, and around 60% for the other muscle models.

# 8 Conclusions and future work

In this paper, the efficiency and accuracy of several muscle modeling alternatives (four muscle models, the number of muscles, muscle torque generators, muscular synergies, and look-up tables for musculotendon lengths and moment arms) has been studied, for gait and vertical jump, in the context of a framework that encompasses the computationally intensive character of predictive simulation algorithms with a known resulting motion that allows establishing a fair comparison among the different alternatives. The framework is a cointegration approach for the forward-dynamics analysis of captured real motions that does not require the unified integration of the multibody and muscle dynamics equations. After the study, the following conclusions can be extracted:

– *Muscle models*. The rigid-tendon model with activation dynamics offers a good compromise between accuracy and efficiency, being especially accurate for short-tendon muscles, as concluded in [20]. The damped-equilibrium model, also tested in this work, does not show any advantage over the full Hill model when used with an implicit integrator, as it was also concluded in [20]. Neglecting activation dynamics or using a nonphysiological model do not provide advantage in efficiency but yield less accurate results. The complexity of the movement seems to have more influence on the error due to the use of simplified models than the fact of being a high- or low-dynamic motion.

– *Number of muscles*. Including muscles in the model leads to a decrease in efficiency, no matter the number of muscles considered. However, such decrease can range from a factor of a few times for a moderate number of muscles and the simplified muscle models, to a factor of one or two orders of magnitude for a large number of muscles and the full Hill model. Run-times scale more than linearly with the number of muscles, with a notably higher growth for the full Hill model than for the simplified models (this means a much more moderate increment in computational cost with the number of muscles than the one obtained in [18] for an optimal control algorithm).

– *Muscle torque generators* (*MTG*). The use of this approach to consider the muscular effect allows to keep the same efficiency as in skeletal models (confirming previous impressions of good efficiency, as that offered by [9]). Although a detailed comparison with the case of modeling the individual muscles cannot be established, the approach yields muscle activation levels that are similar to the weighted average of the activations provided by the muscles contributing to each MTG.

– *Muscular synergies*. The use of muscular synergies limits the capacity of muscles to satisfy the torque constraints of the force-sharing problem, this limitation being reduced as more synergies are considered. Since the optimizer focuses on fulfilling the torque constraints, less effort is put in minimizing the cost function, which reduces the accuracy obtained in activations and forces. However, the efficiency is not significantly improved with respect to the case of not using synergies, which means that the synergy approach is not recommended (the same conclusion obtained in [21] for inverse-dynamics based optimization).

– *Interpolation of configuration-dependent muscular magnitudes*. The use of tables for moment arms, lengths and length derivatives with respect to degrees of freedom does not entail a loss of accuracy, and has a non-negligible impact on the efficiency, especially for the simplified muscle models, reaching gains of around a 60% (less than 25% for the full Hill model as the interpolated terms have a lower relative weight in the corresponding algorithm). However, these gains are much more moderate than those reported in [22] for musculoskeletal kinematics.

**Table 12** Values of the consistency index defined to select the best capture among the set of captures carried out, and values of its three components (kinematic, dynamic, and muscular). For each activity (gait and vertical jump), both the ranges of the four terms and their values for the selected capture are shown

|  | GAIT | | | VERTICAL JUMP | | |
|---|---|---|---|---|---|---|
|  | Max. | Min. | Selected | Max. | Min. | Selected |
| $CI_k$ | 13.30 | 10.62 | 10.96 | 14.82 | 11.46 | 11.46 |
| $CI_d$ | 17.63 | 9.83 | 10.25 | 37.63 | 18.89 | 21.62 |
| $CI_m$ | 61.00 | 26.32 | 61.00 | 52.79 | 19.05 | 52.79 |
| $CI$ | 26.53 | 15.11 | 15.11 | 47.33 | 27.80 | 27.80 |

The approach used in this study, the forward-dynamics analysis of captured real movements, sought to possess the actual characteristics of predictive simulations (iterative, computationally costly), but does not fulfill the essence of a predictive simulation, which is to obtain the unknown motion corresponding to certain inputs. Therefore, an ulterior work must be done, in which the several muscular modeling alternatives are tested in predictive simulation algorithms. While it is likely, but not sure, that the conclusions of this study about efficiency can still be valid in that context, it must be investigated how the differences found in accuracy translate into differences in the resulting motions, and whether they are more or less human-like.

# Appendix

It was said in the paper that the most consistent capture for each movement was selected based on a consistency index that was defined from contributions at kinematic level (minimum dispersion in geometric constraints satisfaction along the motion), at dynamic level (minimum residual in the base body), and at muscular level (best match with EMG signals [21]). In this Appendix, this consistency index is further explained.

At kinematic level, consistency was evaluated by calculating the difference between the position of each marker estimated by the Kalman filter and measured by the optical system. The RMS of these differences, expressed in mm, for all the markers and for all the time points of the movement, was defined as the consistency index for the kinematics, $CI_k$.

At dynamic level, consistency was evaluated by calculating the RMSE between each component of the ground reactions obtained from inverse dynamics and measured by the force plates. The average of the RMSE of the six components was defined as the consistency index for the dynamics, $CI_d$.

At muscular level, consistency was evaluated by calculating the Pearson correlation coefficient $r$ between the obtained muscular activation and the measured EMG for each of the nine muscles where EMG was recorded. The average of the nine coefficients was defined as the consistency index for muscular dynamics, $CI_m$.

Since kinematic and dynamic consistency indices had an order of magnitude of 10, while the Pearson correlation coefficient $r$ varies between $-100$ (no correlation) and $100$ (total correlation), the final consistency index was defined as

$$CI = CI_k + CI_d - \frac{CI_m}{10}. \tag{24}$$

The ranges of the indices obtained in the set of captures carried out for each activity, along with their values for the selected captures, are shown in Table 12.
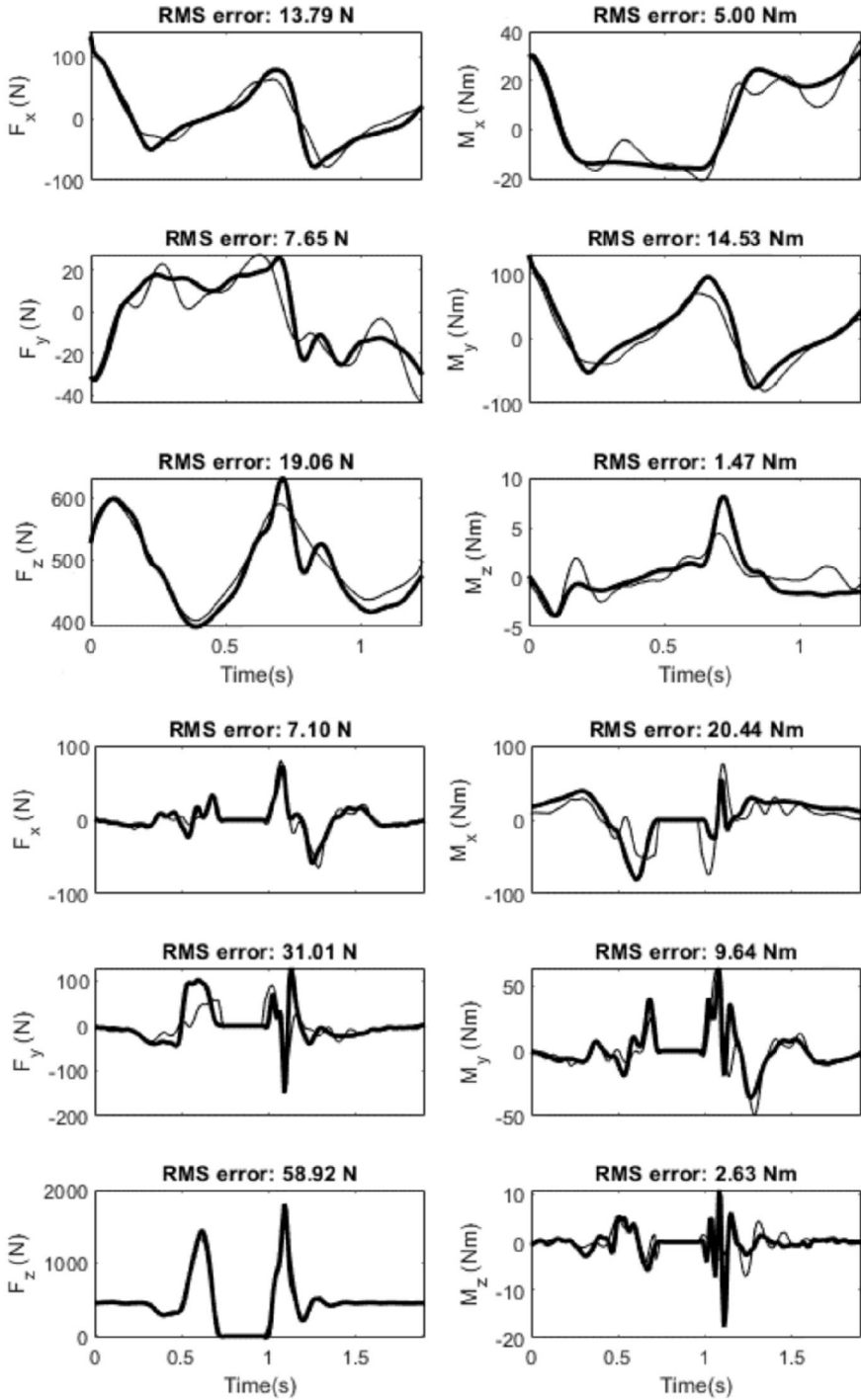
**Fig. 10** Ground reactions obtained from inverse dynamics (thin line) and measured by the force plates (thick line): gait (top); vertical jump (bottom)
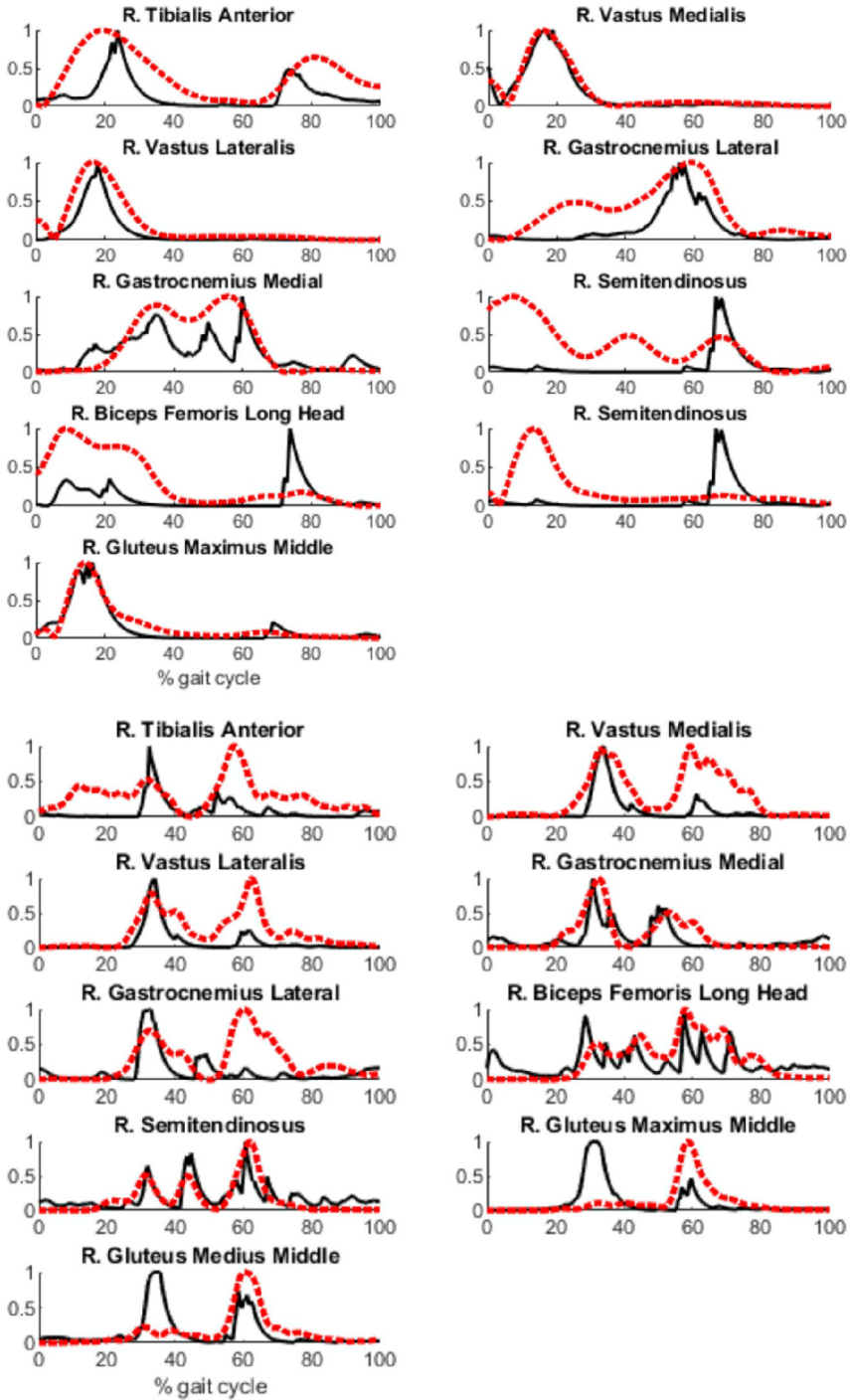
**Fig. 11** Computed muscular activations (solid line) and measured EMGs (dashed line): gait (top); vertical jump (bottom)

Moreover, to provide a better understanding of the meaning of the dynamic and muscular consistency indices, plots of the ground reactions obtained from inverse dynamics and measured by the force plates are provided in Fig. 10, while plots of the computed muscular activations and the measured EMGs are shown in Fig. 11.

# References

1. Ezati, M., Ghannadi, B., McPhee, J.: A review of simulation methods for human movement dynamics with emphasis on gait. Multibody Syst. Dyn. **47**(3), 265–292 (2019)
2. De Groote, F., Falisse, A.: Perspective on musculoskeletal modelling and predictive simulations of human movement to assess the neuromechanics of gait. Proc. R. Soc. Edinb., Sect. B **288**, 20202432 (2021)
3. Ackermann, M., van den Bogert, A.J.: Optimality principles for model-based prediction of human gait. J. Biomech. **43**, 1055–1060 (2010)
4. Manns, P., Sreenivasa, M., Millard, M., Mombaur, K.: Motion optimization and parameter identification for a human and lower back exoskeleton model. IEEE Robot. Autom. Lett. **2**(3), 1564–1570 (2017)
5. Eskinazi, I., Fregly, B.J.: A computational framework for simultaneous estimation of muscle and joint contact forces and body motion using optimization and surrogate modeling. Med. Eng. Phys. **54**, 56–64 (2018)
6. Lin, Y.-C., Walter, J.P., Pandy, M.G.: Predictive simulations of neuromuscular coordination and joint-contact loading in human gait. Ann. Biomed. Eng. **46**, 1216–1227 (2018)
7. Eisentraudt, M., Leyendecker, S.: Epistemic uncertainty in optimal control simulation. In: Mechanical Systems and Signal Processing, vol. 121, pp. 876–889 (2019)
8. Falisse, A., Serrancoli, G., Dembia, C.L., Gillis, J., Jonkers, I., De Groote, F.: Rapid predictive simulations with complex musculoskeletal models suggest that diverse healthy and pathological human gaits can emerge from similar control strategies. J. R. Soc. Interface **16**, 20190402 (2019)
9. Inkol, K.A., Brown, C., McNally, W., Jansen, C., McPhee, J.: Muscle torque generators in multibody dynamic simulations of optimal sports performance. Multibody Syst. Dyn. **50**(4), 435–452 (2020)
10. Dembia, C.L., Bianco, N.A., Falisse, A., Hicks, J.L., Delp, S.L.: OpenSim Moco: musculoskeletal optimal control. PLoS Comput. Biol. **16**(12), e1008493 (2020)
11. Wang, J.M., Hamner, S.R., Delp, S.L., Koltun, V.: Optimizing locomotion controllers using biologically-based actuators and objectives. ACM Trans. Graph. **31**(4), 25 (2012)
12. Geijtenbeek, T., van de Panne, M., van der Stappen, F.: Flexible muscle-based locomotion for bipedal creatures. ACM Trans. Graph. **32**(6), 1–11 (2013)
13. Song, S., Geyer, H.: A neural circuitry that emphasizes spinal feedback generates diverse behaviours of human locomotion. J. Physiol. **593**(16), 3493–3511 (2015)
14. Van der Noot, N., Ijspeert, A.J., Ronsse, R.: Bio-inspired controller achieving forward speed modulation with a 3D bipedal walker. Int. J. Robot. Res. **37**, 168–196 (2018)
15. Geijtenbeek, T.: SCONE: open source software for predictive simulation of biological motion. J. Open Sour. Softw. **4**(38), 1421 (2019)
16. Hämäläinen, P., Eriksson, S., Tanskanen, E., Kyrki, V., Lehtinen, J.: Online motion synthesis using sequential Monte Carlo. ACM Trans. Graph. **33**(4), 51 (2014)
17. Lee, S., Park, M., Lee, K., Lee, J.: Scalable muscle-actuated human simulation and control. ACM Trans. Graph. **38**(4), 1–13 (2019)
18. De Groote, F., Kinney, A.L., Rao, A.V., Fregly, B.J.: Evaluation of direct collocation optimal control problem formulations for solving the muscle redundancy problem. Ann. Biomed. Eng. **44**(10), 2922–2936 (2016)

19. Dorschky, E., Nitschke, M., Seifer, A.K., van den Bogert, A.J., Eskofier, B.M.: Estimation of gait kinematics and kinetics from inertial sensor data using optimal control of musculoskeletal models. J. Biomech. **95**, 109278 (2019)

20. Millard, M., Uchida, T., Seth, A., Delp, S.: Flexing computational muscle: modeling and simulation of musculotendon dynamics. J. Biomech. Eng. **135**(2), 021004 (11 pages) (2013)

21. Michaud, F., Lamas, M., Lugris, U., Cuadrado, J.: A fair and EMG-validated comparison of recruitment criteria, musculotendon models and muscle coordination strategies, for the inverse-dynamics based optimization of muscle forces during gait. J. NeuroEng. Rehabil. **18**, 17 (2021)

22. Sobinov, A., Boots, M.T., Gritsenko, V., Fisher, L.E., Gaunt, R.A., Yakovenko, S.: Approximating complex musculoskeletal biomechanics using multidimensional autogenerating polynomials. PLoS Comput. Biol. **16**(12), e1008350 (2020)

23. Mouzo, F., Lugris, U., Pamies-Vila, R., Cuadrado, J.: Skeletal-level control-based forward dynamic analysis of acquired healthy and assisted gait motion. Multibody Syst. Dyn. **44**(1), 1–29 (2018)

24. Thelen, D.G., Anderson, F.C.: Using computed muscle control to generate forward dynamic simulations of human walking from experimental data. J. Biomech. **39**, 1107–1115 (2006)

25. Cuadrado, J., Dopico, D., Gonzalez, M., Naya, M.A.: A combined penalty and recursive real-time formulation for multibody dynamics. J. Mech. Des. **126**(4), 602–608 (2004)

26. Cuadrado, J., Michaud, F., Lugrís, U., Perez Soto, M.: Using accelerometer data to tune the parameters of an extended Kalman filter for optical motion capture: preliminary application to gait analysis. Sensors **21**(2), 427 (2021)

27. Garcia de Jalon, J., Bayo, E.: Kinematic and dynamic simulation of multibody systems. Springer, Berlin (1994)

28. Dopico, D.: MBSLIM: multibody systems en laboratorio de ingenieria mecanica (2016). http://lim.ii. udc.es/MBSLIM

29. Lugris, U., Carlin, J., Luaces, A., Cuadrado, J.: Gait analysis system for spinal cord injured subjects assisted by active orthoses and crutches. J. Multi-Body Dyn. **227**(4), 363–374 (2013)

30. Michaud, F., Lamas, M., Lugris, U., Cuadrado, J.: A fair and EMG-validated comparison of recruitment criteria, musculotendon models and muscle coordination strategies, for the inverse-dynamics based optimization of muscle forces during gait. J. NeuroEng. Rehabil. **18**, 17 (2021)

31. Romero, F., Alonso, F.J., Gragera, C., Lugris, U., Font-Llagunes, J.M.: Estimation of muscular forces from SSA smoothed sEMG signals calibrated by inverse dynamics-based physiological static optimization. J. Braz. Soc. Mech. Sci. Eng. **38**(8), 2213–2223 (2016)

32. Delp, S.L., Anderson, F.C., Arnold, A.S., Loan, P., Habib, A., John, C.T., Guendelman, E., Thelen, D.G.: OpenSim: open-source software to create and analyze dynamic simulations of movement. IEEE Trans. Biomed. Eng. **54**(11), 1940–1950 (2007)

33. Zajac, F.E.: Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. Crit. Rev. Biomed. Eng. **17**(4), 359–411 (1989)

34. Michaud, F.: Neuromusculoskeletal human multibody models for the gait of healthy and spinal-cord-injured subjects. PhD Dissertation, University of La Coruña, Ferrol (2020)

35. Anderson, D.E., Madigan, M.L., Nussbaum, M.A.: Maximum voluntary joint torque as a function of joint angle and angular velocity: model development and application to the lower limb. J. Biomech. **40**(14), 3105–3113 (2007)

36. Scholz, A., Sherman, M., Stavness, I., Delp, S., Kecskemethy, A.: A fast multi-obstacle muscle wrapping method using natural geodesic variations. Multibody Syst. Dyn. **36**(2), 195–219 (2016)