Lower limb stiffness estimation during running: the effect of using kinematic constraints in muscle force optimization algorithms

Roberto Bortoletto[1], Enrico Pagello[1], and Davide Piovesan[2]

[1] Intelligent Autonomous Systems Laboratory (IAS-Lab), University of Padua
Department of Information Engineering, Via G. Gradenigo 6/b, 35131, Padova, Italy.
{roberto.bortoletto,epv}@dei.unipd.it
[2] Biomedical Program, Mechanical Engineering Department
Gannon University, 109 University Square, PMB #3251, Erie, PA 16541, USA
piovesan001@gannon.edu

Abstract. The focus of this paper is on the effect of muscle force optimization algorithms on the human lower limb stiffness estimation. By using a forward dynamic neuromusculoskeletal model coupled with a muscle short-range stiffness model we computed the human joint stiffness of the lower limb during running. The joint stiffness values are calculated using two different muscle force optimization procedures, namely: Toque-based and Torque/Kinematic-based algorithm. A comparison between the processed EMG signal and the corresponding estimated muscle forces with the two optimization algorithms is provided. We found that the two stiffness estimates are strongly influenced by the adopted algorithm. We observed different magnitude and timing of both the estimated muscle forces and joint stiffness time profile with respect to each gait phase, as function of the optimization algorithm used.

Keywords: joint stiffness, muscle force, optimization algorithms

1 Introduction

During the last two decades the interest in understanding the physiological basis of human and animal movement has resulted in an extensive range of experiments. The study of human movement has been improved through the introduction of muscle-driven dynamic simulations. This approach includes mathematical models of muscle activation and contraction dynamics and allows for the calculation of muscle forces, fiber lengths, and other parameters that cannot be easily measured in-vivo. Muscle-driven simulations have been used in a wide variety of applications, including the analysis of human walking [1–3], running [4], and pathological gait [5]. Biomechanical models have been used in several studies to predict the muscle forces and joint torques along with human body motion. One of the first muscle's mathematical models was proposed by Hill [6]. Gordon et al. [7] refined such model by incorporating the dependence between changes in muscle force as function of muscle lengths and contraction speeds. Zajac extended

the Hill's model introducing a muscle-tendon model [8], which is known as Hill-type muscle force model. The prediction of muscle force can be also calculated independently from a model by means of optimization algorithms. Such algorithms are usually based on a cost function that depends directly on a physical parameters such as force variance, energy, muscle stress, to name a few [9–11]. When accomplishing a task, that require both the following of a trajectory and the exertion of a force, humans need to modulate not only the generated muscle forces, but also the corresponding limb stiffness. During unimpaired gait, depending on the terrain, one might either walk in a relaxed manner or stiffen up to increase stability. Several studies have been proposed to estimate the hip [12], knee [13, 14] and ankle [15] joint stiffness to characterize the mechanical properties of the whole limb [16–18]. Moreover, a preliminary study about the incidence of the adopted muscle-tendon model on the joint stiffness estimation has been proposed in [19].
This paper targets two main research topics: the use of forward dynamic neuromusculoskeletal modeling to estimate muscle forces, joint moments, and joint kinematics from biological signal, and the use of muscle short-range stiffness to estimate human lower limb joint stiffness. In particular, the aim of this study is to evaluate how the adoption of different muscle force optimization algorithms, based on the inclusion or exclusion of kinematic constraints, affects the lower limb stiffness estimation during running.

2 Methods

In this study we performed a series of simulations that coupled a 3D human musculoskeletal model of the lower limb with a model of muscle stiffness, to estimate leg's joint stiffness during running. The algorithm to compute joint stiffness is dependent on the value of the computed muscle force. We compared the stiffness values obtained in simulation using two different muscle force optimization procedures, that we call *Torque-based Muscle Force computation* and *Torque/Kinematic-based Muscle Force computation*. The musculoskeletal model that we used is freely available with the OpenSim[3] platform. It includes seven body segments for each leg: pelvis, femur, patella, tibia-fibula, talus, foot, and toes. Each foot includes the calcaneus, navicular, cuboid, cuneiformis, and metatarsal. Each arm is represented by humerus, ulna, radius, and hand. The joint definitions are derived by [20, 21] and the anthropometry by [22]. The physiological parameters of muscles are in accordance to mean values reported in [23]. 92 muscle-tendon actuators represent the main muscle groups: 43 for each leg, and 6 lumbar muscles. The arm joints are actuated by ideal torque motors. The experimental dataset used here is freely available within the "*Muscle function of overground running across a range of speeds*"[4] section of the SimTk.org[5]. The website is a public repository for data, models, and computational tools related

[3] OpenSim Project Overview: https://simtk.org/home/opensim
[4] Project's page: https://simtk.org/home/runningspeeds
[5] Repository's home page: https://simtk.org/xml/index.xml

to physics-based simulation of biological systems. These data were originally collected with the purpose to better understand how the leg muscles coordinate motion of the body segments during running. A detailed description of the adopted experimental protocol is available in [24]. We considered the Electromyographic (EMG), Motion Capture (MC) and Ground Reaction Forces (GRFs) data referred to a male subject (age, 19 years; mass, 75.9 Kg; height, 1.82 m; leg length, 1.00 m) that at the time of testing were not suffering from any musculoskeletal injury likely to adversely affect their sprinting ability. Among the data we considered a medium-paced running speed at 5.20 ms^{-1}. The raw data, available in .c3d format[6], were processed in Matlab to extract the information relative to the kinematic, the GRFs and the EMG signals of the subject. The kinematic and GRFs data were used in OpenSim to scale the musculoskeletal model to the anthropometry of the real subject, and to compute the joint angle values by solving the Inverse Kinematics (IK) problem. Then, using the Residual Reduction Algorithm[7] (RRA) we optimized the model adjusting the mass distribution and joint kinematics to make them consistent with GRFs. The next step was to solve the Inverse Dynamics (ID) problem to compute the joint torques. At this point, the force generated by each muscle was estimated by adopting the two optimization techniques: *Torque-based Muscle Force computation* (subsection 2.1) and *Torque/Kinematic-based Muscle Force computation* (subsection 2.2), which do not involve the use of a specific muscle model, but only information about the Maximum Isometric Force (MIF) exertable by each muscle. The waveforms of the EMG signals were used to highlight the similarities and differences between the obtained estimates, compared with the experimental data.

2.1 Torque-based Muscle Force computation

The *Torque-based Muscle Force computation* procedure is based on the use of an optimization which estimates the distribution of muscle forces for a specific set of joint torques. The cost function is the sum of the squared muscle forces (Eq.1), expressed as a fraction of the MIF for each muscle. A set of constraints was considered such that the resulting muscle forces summed to the specific joint torque (Eq.2), and muscle forces were positive and less than or equal to the achievable MIF (Eq.3).

$$u = \min \sum_{i=1}^{M} \left(\frac{F_{m,i}}{F^0_{m,i}} \right)^2 \tag{1}$$

$$\tau_j = \sum_{i=1}^{M} r_{ij} \times F_{m,i} \qquad i = 1, 2, ..., M; \quad j = 1, 2, ..., N \tag{2}$$

$$0 \leq F_{m,i} \leq F^0_{m,i} \tag{3}$$

[6] C3D File Format Specification: http://www.c3d.org
[7] http://simtk-confluence.stanford.edu:8080/display/OpenSim/How+RRA+Works

where $F_{m,i}$ is the muscle force of the i-th muscle and $F^0_{m,i}$ is the corresponding MIF, r_{ij} is the posture-dependent moment arm for the i-th muscle relative to the j-th joint, and τ_j is the torque about the j-th joint. The sum over M elements corresponds to the number of muscle-tendon actuator crossing the hip, knee and ankle joint in the model. N is the number of Degrees of Freedom (DOFs). The data related to $F^0_{m,i}$, r_{ij}, and τ_j were taken from the musculoskeletal model and from the simulation executed within OpenSim. In which the ID problem was solved taking into account the GRFs. Given these data as input and minimizing the cost function with respect to the constraints we obtained the corresponding muscle forces $F_{m,i}$. The *Torque-based Muscle Force computation* procedure was performed in Matlab environment.

2.2 Torque/Kinematic-based Muscle Force computation

The *Torque/Kinematic-based Muscle Force computation* algorithm represents the procedure available within OpenSim to compute the muscle excitation and the corresponding muscle force. It is based on the use of the Computed Muscle Control (CMC) tool [25]. We extracted the information related to the muscle force by executing the CMC in order to compute the muscle excitation levels that drives the generalized coordinates of the dynamic musculoskeletal model towards the desired trajectory. The available implementation is based on the combination of Proportional-Derivative (PD) control and Static Optimization (SO) that allow to conduct a standard forward dynamic simulation. A detailed description of the CMC tool operating principles is available in [26]. We report here only the main concepts with particular reference to the computation of the muscle forces. In this study, the SO toolbox is used by CMC to resolve the net joint torques and moments into individual muscle forces subject to the following muscle activation-to-force condition:

$$\sum_{i=1}^{M} a_{m,i} F^0_{m,i} r_{ij} = \tau_j \quad \forall j \tag{4}$$

where M is the number of muscles, $a_{m,i}$ is the activation level ($0 < a_{m,i} \leq 1$), $F^0_{m,i}$ is the MIF, and r_{ij} is the moment arm of the i-th muscle about the j-th joint. τ_j is the joint torque acting about the j-th joint. In CMC, an objective function J combines the sum of squared muscle activations augmented by a set of equality constraints ($C_j = 0$) that requires the desired accelerations to be achieved within the tolerance set for the optimizer (Eq.5).

$$J = \sum_{i=1}^{M} \left(a_{m,i}\right)^2; \qquad C_j = \ddot{q}^*_j - \ddot{q}_j \quad \forall j \tag{5}$$

where M is the number of muscles and $a_{m,i}$ is the activation level ($0 < a_{m,i} \leq 1$). This setup is equivalent to having an ideal force generators in which the model accelerations $\ddot{q}_j$ are driven toward the desired accelerations $\ddot{q}^*_j$, where q_j

respresents the j-th model coordinate. The desired accelerations are computed using the following PD control law:

$$\ddot{\boldsymbol{q}}^*(t+T) = \ddot{\boldsymbol{q}}_{exp}(t+T) + \boldsymbol{k}_v[\dot{\boldsymbol{q}}_{exp}(t) - \dot{\boldsymbol{q}}(t)] + \boldsymbol{k}_p[\boldsymbol{q}_{exp}(t) - \boldsymbol{q}(t)] \tag{6}$$

where $\boldsymbol{q}_{exp}$ are the experimentally-derived coordinates, $\ddot{\boldsymbol{q}}^*$ are the desired accelerations, and $\boldsymbol{q}$ are the model coordinates. $\boldsymbol{k}_v = 30$ and $\boldsymbol{k}_p = 900$ are the feedback gains on velocity and position errors, which were experimentally set.
It is worth noting that, by assuming $0 \leq F_{m,i} \leq F^0_{m,i}$ in *Torque-based Muscle Force computation* and $0 < a_{m,i} \leq 1$ in *Torque/Kinematic-based Muscle Force computation*, then Eq.4 is equivalent to Eq.2, and Eq.5 is equivalent to Eq.1. This because, the ratio $F_{m,i}/F^0_{m,i}$ always varies between 0 and 1. On the other hand, the introduction of the kinematic constraints ($C_j = \ddot{q}^*_j - \ddot{q}_j \quad \forall j$) in Eq.5, encompassing the coordinate accelerations, significantly diversifies the two approaches considered here. This is especially true since we are analyzing dynamic movement such as running in its different phases and not simply static poses.

2.3 Stiffness Estimation

The adopted muscle model is known in literature as *Thelen2003Muscle* [27], which is the name of its implementation within the OpenSim. The model is based on the Hill-type muscle model, which represents a muscle-tendon unit through three elements: a contractile element (CE), a parallel element (PE), and a series element (SE). While CE is responsible for the active force, PE accounts for the muscle passive behavior and SE represents the tendon. *Thelen2003Muscle* is a parametric-based implementation capable of representing the muscle mechanical response by considering the following physiological parameters: MIF, Optimal Muscle Fiber Length (OMFL), Tendon Slack Length (TSL), Maximum Contraction Velocity (MCV), and Pennation Angle (PA). Given the normalized muscle force obtained either within CMC or with our dedicated algorithm, the muscle fiber length was estimated by means of Force-Length relationship for each muscle embedded in the OpenSim's Muscle model. The force-length curves modeled the effects of active/passive muscle components. In particular, the active force-length curve was described using natural cubic splines [8], while the passive force-length curve was described using exponential functions [27]. The tendon length was estimated by subtracting the fiber length to the whole-muscle length derived from the path geometry and joint angles within the simulation.
Finally, the short-range muscle stiffness was estimated using the model developed by Cui et al. [28], and already adopted by Perreault et al. [29] for the estimation of the endpoint stiffness of the human arm. The model assumes that the short-range stiffness of a muscle-tendon unit, K_{mt} (Eq.7), results from the stiffness of the muscle fibers, K_m, in series with the stiffness of the tendon, K_t.

$$K_{mt} = \frac{K_m K_t}{K_m + K_t} \tag{7}$$

K_m is a function of the muscle force F_m and OMFL l^m_0, up to a dimensionless scaling constant. K_t is defined by the ratio between the tendon force and the

tendon elongation, which is given by the difference between the tendon length l_t and the TSL l_s^t. By using the estimated muscle forces and the muscle short-range stiffness, we computed the corresponding joint stiffness taking into account the kinematic relationship between changes in joint angles and changes in muscle-tendon length (Eq.8).

$$K_j = J^T \tilde{K}_{mt} J + \frac{\partial J^T}{\partial \theta} \tilde{F}_m \tag{8}$$

where J is the Jacobian matrix relating changes in muscle joint angles to changes in muscle length, $\tilde{K}_{mt}$ is a diagonal matrix with the stiffness for each muscle in the model, $\tilde{F}_m$ is the vector of muscle forces, and θ is the vector of joint angles. The partial derivative of the Jacobian matrix with respect to joint angles accounts for how angle dependent changes in muscle moment arms influence joint stiffness. It is worth noting that K_j is a 3×3 matrix in which the diagonal elements represent the hip, knee and ankle joint stiffness, while the other elements represent the stiffness relationship existing between hip and knee, hip and ankle, knee and ankle. In particular, not having tri-articular muscles, in this study the hip-ankle stiffness relationship is zero. A comparison between the obtained joint stiffness using the two different muscle force optimization algorithms has been considered to show how the choice of a specific algorithm, for the estimation of force, affects the result we get in the computation of stiffness.

3 Results

The set of results, provided here, shows significant differences in the muscle force estimates obtained by adopting either the *Torque-based Muscle Force computation* or the *Torque/Kinematic-based Muscle Force computation* procedure. A first evaluation has been done by performing a cross-correlation analysis between muscle forces estimated with both *Torque-based Muscle Force computation* and *Torque/Kinematic-based Muscle Force computation* procedures, and the processed EMG signal (Fig. 1). The obtained cross-correlation sequences were normalized so that the autocorrelations at zero lag were identically 1.0. The results show that, for the muscles most involved in the analyzed movement (i.e. Gastrocnemius, Vasti, Rectus Femoris and Bicep Femoris), the cross-correlation related to the *Torque-based Muscle Force computation* is higher than that obtained for the *Torque/Kinematic-based Muscle Force computation* procedure.
Fig. 2 shows a comparison between the processed EMG signal of such representative muscles and their estimated forces. It is clear that most of the time the *Torque-based Muscle Force computation* (blue line) tracks EMG signal better than the *Torque/Kinematic-based Muscle Force computation* (green line), which gives large forces also when there is not EMG signal. This makes the latter less likely to be appropriate. During the optimization stage, through which the muscle force is computed, it may also happen that a high value of the EMG signal does not correspond to an equally high developed muscle force (Fig. 2, Right Rectus Femoris). This behavior may be due to a configuration of the muscle for which, despite the activation level, the fibers length is either very

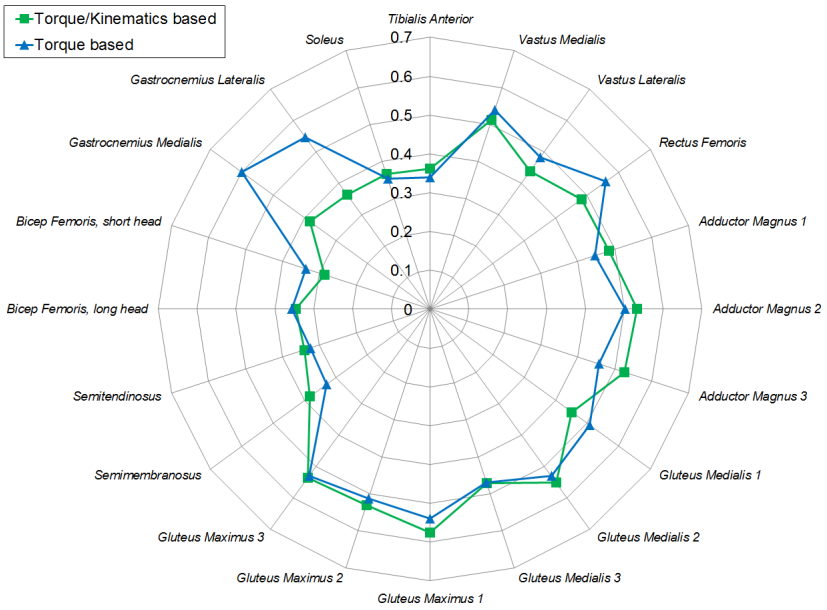

Fig. 1. Maximum values of the cross-correlation functions computed between muscle forces estimated with both *Torque-based Muscle Force computation* and *Torque/Kinematic-based Muscle Force computation* procedures, and the processed EMG signal.

stretched or very contracted and does not allow the generation of an appropriate force with respect to the OMFL. Recall that the MIF is generated only at the OMFL. Future study should take into consideration methods to estimate muscle forces that better track the corresponding EMG profiles. Corresponding differences also arise in the stiffness values obtained by adopting either *Torque-based Muscle Force computation* or *Torque/Kinematic-based Muscle Force computation* procedure. As depicted in Fig.3 there is a misalignment of the peaks of the stiffness time profiles with respect to the different phases of movement. We can notice a delay between the instants in which the foot impacts the ground and the instant in which the stiffness peaks generated by either approach occurs. Notice that *Torque/Kinematic-based Muscle Force computation* has an average delay of 112 ms compared to the 82 ms of *Torque-based Muscle Force computation*. Furthermore, the former produces stiffness peaks with a much larger amplitude compared to the latter. The stiffness peaks for both knee and ankle occurs almost synchronously within each model. Furthermore, we can notice that the width of stiffness peaks are different between algorithms. For example, the graph of the knee stiffness shows that the knee is contracted for a longer time in the *Torque/Kinematic-based Muscle Force computation*. The ratio of hip/knee stiffness at the peak is different for the two approaches: 1.28 (*Torque-based Muscle Force computation*), 1.86 (*Torque/Kinematic-based Muscle Force computation*).

The second ratio is bigger indicating a predominancy of hip stiffness over the other joints. Similar considerations hold true for the left lower limb joints.

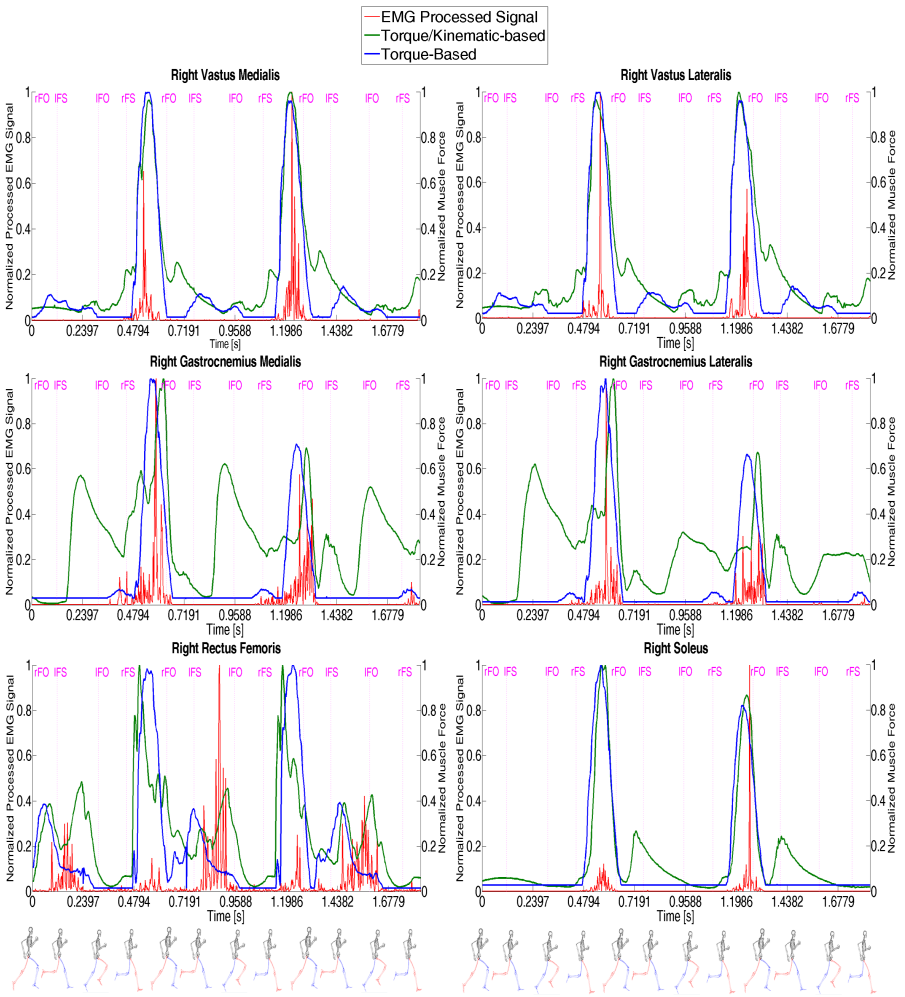

Fig. 2. Processed EMG signal profiles compared to estimated muscle forces with *Torque-based Muscle Force computation* and *Torque/Kinematic-based Muscle Force computation* procedures. The labels stand for Left Foot-Off (lFO), Left Foot-Strike (lFS), Right Foot-Off (rFO), and Right Foot-Strike (rFS).

The inter-joint stiffness estimated in this work for each combination of two joints was found to be symmetric. Furthermore the inter-joint stiffness between hip and ankle was negligible. Thus, only two inter-joint stiffness time profiles are shown in Fig. 4. These results are an indication that all the algorithms were implemented

properly. Indeed, since the stiffness is a positive definite tensor it is expected to be simmetric. Moreover, due to the absence of tri-articular muscles connecting the ilium with the foot the hip-ankle component must be zero.

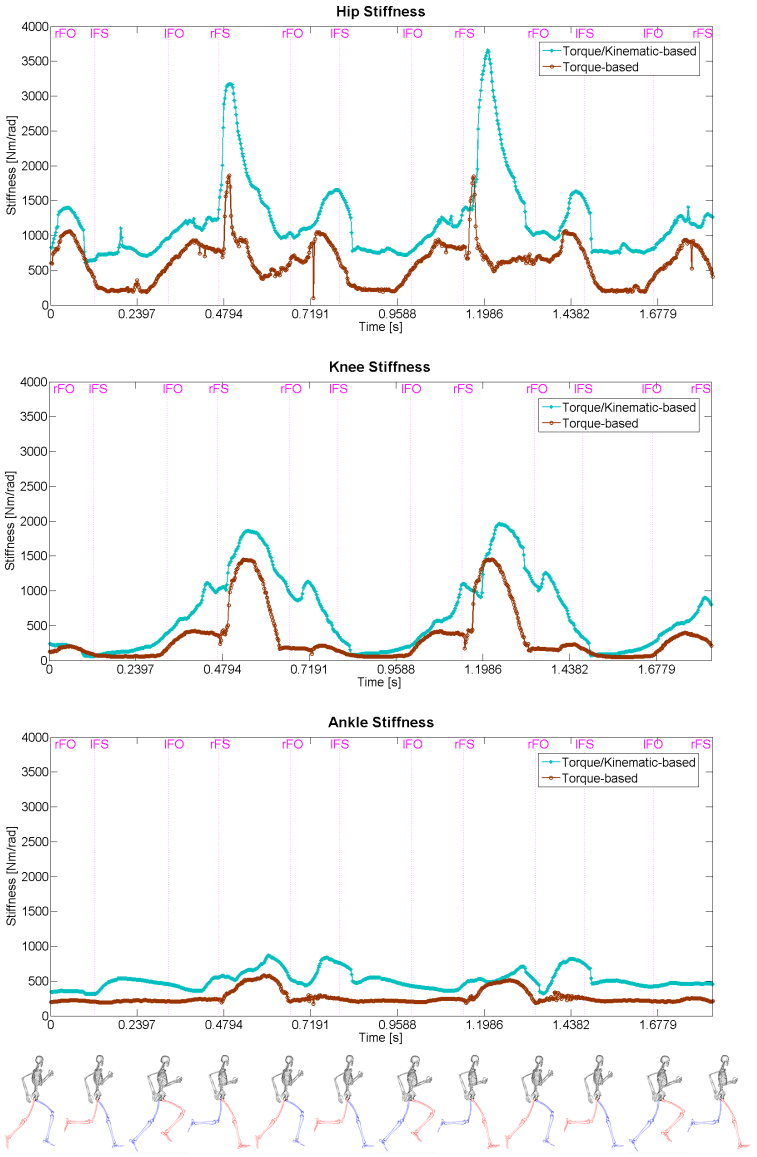

Fig. 3. Right Lower Limb Joint Stiffness estimated values: Hip, Knee, and Ankle. The x-axis reports the time-samples, while y-axis expresses the joint stiffness [Nm/rad]. The labels stand for Left Foot-Off (lFO), Left Foot-Strike (lFS), Right Foot-Off (rFO), and Right Foot-Strike (rFS).

The inter-joint stiffness is not negligible for the pair ankle-knee and knee-hip. However, the magnitude of these stiffness time-profiles is smaller than those proper of the joints.

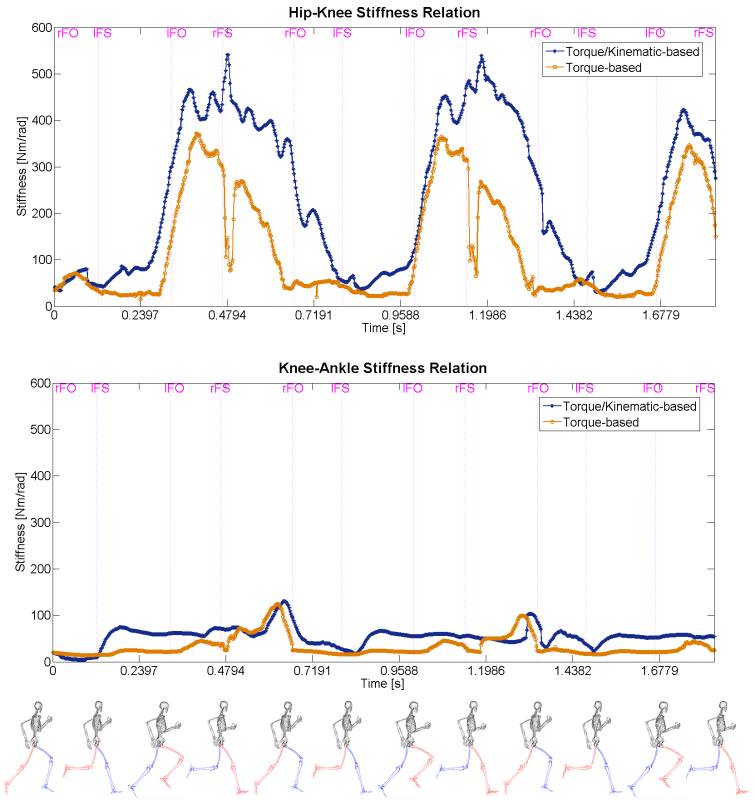

Fig. 4. Right Lower Limb Intra-Joint Stiffness estimated values: Hip-Knee Stiffness Relation, and Knee-Ankle Stiffness Relation. The x-axis reports the time-samples, while y-axis expresses the intra-joint stiffness [Nm/rad]. The labels stand for Left Foot-Off (lFO), Left Foot-Strike (lFS), Right Foot-Off (rFO), and Right Foot-Strike (rFS).

4 Conclusions

In this paper, two different whole-muscle force optimization algorithms are utilized to estimate the lower limb muscle-tendon forces and the corresponding joint stiffness during the running of an unimpaired individuals. It is important to note that the purpose of this study was not to determine which algorithm is better for the estimation of muscle forces, but the goal was to determine how different algorithms may affect the estimation of joint stiffness. Indeed, results

show that the choice of the optimization algorithm influences the estimation of the muscle-tendon stiffness and of the corresponding joint stiffness. The adopted modeling and simulation techniques highlight how it is possible to estimate the joint stiffness decomposing the computation into two stages, where the assumption of a muscle-tendon model is actually required only in the computation of the geometrical parameters such as the muscle lengths and moment arms. At the same time, there are a number of open questions related to the possibility of estimating the stiffness during the execution of the movement and not only in relation to particular limb poses. Further studies are needed in order to provide a more precise modeling of the muscle-tendon unit capable of describing how the behavior and the parameterization of the muscle-tendon unit changes as a function of the posture. Future research will focus on providing a better characterization of the existing relationships between muscle models, muscle-tendon force optimization algorithms and stiffness estimation procedures.

Acknowledgements

This research has been supported by Consorzio Ethics through a grant for research activity on the project Rehabilitation Robotics, and by the Faculty research grant at Gannon University.

References

1. Anderson, F. C., and Pandy, M. G. Dynamic Optimization of Human Walking. ASME J. Biomech. Eng., 123(5), pp. 381-390.
2. Ackerman, M., and van der Bogert, A. J. Optimality Principles for Model-Based Prediction of Human Gait. J. Biomech., 43(6), pp. 1055-1060.
3. Arnold, E. M., and Delp, S. L. Fibre Operating Lengths of Human Lower Limb Muscle During Walking. Philos. T. R. Soc. B., 366(1570), pp. 1530-1539.
4. Hamner, S. R., Seth, A., and Delp, S. L. Muscle Contributions to Propulsion and Support During Running. J. Biomech., 43(14), pp. 2709-2716.
5. Steele, K. M., Seth, A., Hicks, J. L., Schwartz, M. S., and Delp, S. L. Muscle Contributions to Support and Progression During Single-Limb Stance in Crouch Gait. J. Biomech. 43(11), pp. 2099-2105.
6. Hill, A. V. The Heat of Shortening and the Dynamic Constants of Muscle. Proc. R. Soc. Lond. B. 1938.
7. Gordon, A. M., Huxley, A. F., and Julian, F. J. The variation in isometric tension with sarcomere length in vertebrate muscle fibres. J. of Phys, 184:170-192. 1966.
8. Zajac, F. E. Muscle and tendon: properties, models, scaling, and application to biomechanics and motor contro. Crit. Rev. Biomed. Eng. 17, pp. 359-411. 1989.
9. Anderson, F. C., Pandy, M. G. Static and dynamic optimization solutions for gait are pratically equivalent. J. Biomech., 34(2), pp. 153-161.2001.
10. Erdemir, A., McLean, S., Herzog, W., van den Bogert, A. J. Model-based estimation of muscle forces exerted during movements. Clinical Biomechanics 22, pp. 131-154. 2007.
11. Monaco, V., Coscia, M., Micera S. Cost function tuning improves muscle force estimation computed by static optimization during walking. Conf. Proc. IEEE Eng. Med. Biol. Soc., 2011.

12. Shamaei, K. and Sawicki, G. S. and Dollar, A. M. Estimation of Quasi-Stiffness of the Human Hip in the Stance Phase of Walking. PloS one, Vol. 8, No. 12. 2013.
13. Pfeifer, S. and Vallery, H. and Hardegger, M. and Riener, R. and Perreault, E. J. Model-based estimation of knee stiffness. IEEE trans. on bio-medical engineering, Vol. 59, No. 9, pp. 2604-2615. 2012.
14. Shamaei, K. and Sawicki, G. S. and Dollar, A. M. Estimation of quasi-stiffness of the human knee in the stance phase of walking. PloS one, Vol. 8, No. 3. 2013.
15. Shamaei, K. and Sawicki, G. S. and Dollar, A. M. Estimation of quasi-stiffness and propulsive work of the human ankle in the stance phase of walking. PloS one, Vol. 8, No. 3. 2013.
16. Piovesan D., Pierobon A., DiZio P., Lackner J.R., Experimental Measure of Arm Stiffness During Single Reaching Movements with a Time-Frequency Analysis. J.Neurophysiol. 2013; 110(10), 2484-2496.
17. Piovesan D., Casadio M., Morasso P., Giannoni P., Arm stiffness during assisted movements following stroke: the influence of visual feedback and training. IEEE Trans Neural Syst Rehabil Eng. 2013; 21(3), 454-465.
18. Piovesan D., Pierobon A., DiZio P., Lackner J.R., Measuring Multi-Joint Stiffness during Single Movements: Numerical Validation of a Novel Time-Frequency Approach. PLoS ONE 2012; 7(3): e33086.
19. Bortoletto, R., Pagello, E., Piovesan, D. How different human muscle models affect the estimation of lower limb joint stiffness during running. Accepted for publication in Proc. of Workshop on Neuro-Robotics for Patient-Specific Rehabilitation, July 18, 2014. IAS-13 Conf., July 15-19, Padua, 2014.
20. Yamaguchi, G. T., and Zajac, F. E. A planar model of the knee joint to characterize the knee extensor mechanism. J. Biomech 22, 1-10, 1989.
21. Delp, S. L., Loan, J. P., Hoy, M. G., Zajac, F. E., Topp, E. L., and Rosen, J. M. An Interactive Graphics-Based Model of the Lower Extremity to Study Orthopaedic Surgical Procedures. IEEE Trans. Biomed. Eng., 37(8), pp. 757-767.
22. Anderson, F. C., Pandy, M. G. A Dynamic Optimization Solution for Vertical Jumping in Three Dimensions. Comput Methods Biomech Biomed Engin. 1999. January, 2(3): 201-31.
23. Ward, S. R., Eng, C. M., Smallwood, L. H., and Lieber, R. L. Are current measurements of lower extremity muscle architecture accurate? Clin. Orthop. Relat. Res. 467, 1074-1082. 2009.
24. Dorn, T. W., Schache A. G., and Pandy, M. G. Muscular startegy shift in human running: dependence of running speed on hip and ankle muscle performance. The J. of Exp. Biol., 215, 1944-1956. 2012.
25. Thelen, D. G., Anderson, F. C., Delp, S. L. Generating dynamic simulations of movement using computed muscle control. J. of Biomec 36 (2003), pp.321-328.
26. Thelen, D. G., Anderson, F. C. Using computed muscle control to generate forward dynamic simulations of human walking from experimental data. J. Biomech. 2006, January 39(6): 1107-1115.
27. Thelen D. G., Adjustment of Muscle Mechanics Model Parameters to Simulate Dynamic Contractions in Older Adults. J. Biomech Eng, 2003, 125(1):70.
28. Cui L., Perreault, E. J., Maas, H. and Sandercock T. G. Modeling short-range stiffness of feline lower hindlimb muscles. J. Biomech 41: 1945-1952, 2008.
29. Hu, X. and Murray, W. M. and Perreault, E. J. Muscle short-range stiffness can be used to estimate the endpoint stiffness of the human arm. Journal of neurophysiology. Vol. 105, No. 4, pp. 1633-1641, 2011.