# 7

# Knee joint modeling using OpenSim software



Several computational codes such as SIMPACK (Rulka, 1990), AnyBody Modeling System™ (AnyBody Technology, Aalborg, Denmark), MADYMO® (Tass, Rijswijk, Netherlands), PC Crash™ (MEA Forensic, Vancouver, Canada), LifeModeler™ (LifeModeler, Inc., San Clemente, CA), SIMM (Delp *et al.*, 1990), APOLLO (Silva, 2003) and OpenSim (Delp *et al.*, 2007) in a form of academic and/or commercial packages have been available for biomechanical modeling and analysis. These computational tools use multibody system (MBS) methodologies and permit to carry out several kinematic and dynamic analyses. Within these analysis tools, a wide range of internal human variables can be monitored, such as (*i*) kinematic data, i.e. positions, velocities and accelerations; (*ii*) dynamic musculoskeletal data, i.e. muscle lengths, muscle forces, joint reactions, etc.; or (*iii*) metabolic power consumption (Agarwal *et al.*, 2010). Furthermore, prostheses and orthoses can be included into computational models of the human body with the intent to get some insights about the design and dynamic performance of such medical devices and walking aids (Lin *et al.*, 2010; Mihalko *et al.*, 2012; Dao *et al.*, 2012).

In this Chapter, the capabilities of OpenSim software for modeling and analyzing the kinematics and dynamics of musculoskeletal systems are investigated. A modeling framework explaining how to build a multibody knee model in OpenSim is presented, being the limitations of using this software pointed out. This framework for contact and muscle force modeling comprises four main tasks: (*i*) develop an OpenSim model of the knee joint, (*ii*) perform inverse kinematic analysis to find the set of generalized coordinates for the model that best match the experimental kinematics recorded for a particular subject, (*iii*) introduce the main muscles and associated tendons responsible for the desired kinematics into the model and (*iv*) define the knee articular surfaces and a contact model to compute the contact forces.

## 7.1 Overview of OpenSim software

OpenSim is an open-source software developed under the framework of MBS methodologies that (*i*) allows for the construction and simulation of musculoskeletal models, (*ii*) permits the visualization of experimental and simulated motion, and (*iii*) provides a set of analysis tools, such as inverse kinematics, inverse dynamics, static optimization, forward dynamics, computed muscle control (CMC), etc. A scaling tool is also offered for generating subject-specific simulation (Seth *et al.*, 2011).

The OpenSim provides an online platform, in which the biomechanics community can release new projects, libraries, models, tutorials, etc. This platform allows for testing, analyzing and enhacing models from several sources and, therefore, promotes multi-institutional collaborations. OpenSim includes two interfaces: an end-user application with a graphical user interface (GUI) and an application programming interface (API) that allows researchers to access and customize OpenSim core functionality. The core software is written in C++, and the graphical user interface (GUI) is written in Java. OpenSim plug-in technology makes possible to develop customized controllers, analyses, contact models and muscle models. These plug-ins can be shared without the need to change or compile source code. OpenSim is organized into computational and functional layers, as it is illustrated in Figure 7.1. The base layer is the Simbody$^{TM}$ library (Delp *et al.*, 1990), an open-source order-*n* dynamics engine developed for creating mathematical models of biological dynamics. The next layer is the modeling layer, which comprises two units: ModelComponent and Model. The ModelComponent unit is applied to represent the physical parts of a model, namely

bodies, joints, constraints, forces, actuators and controllers. The Model unit is aimed to assemble all these physical parts of the model in order to get a coherent and consistent system. The modeling layer is followed by the analysis layer, which falls into three categories: modeler, solver and reporter. The top layer is the application layer that contains the OpenSim GUI or dynamic libraries encapsulating model components and/or analyses that are performed in the GUI (Anderson *et al.*, 2011, Seth *et al.*, 2011).

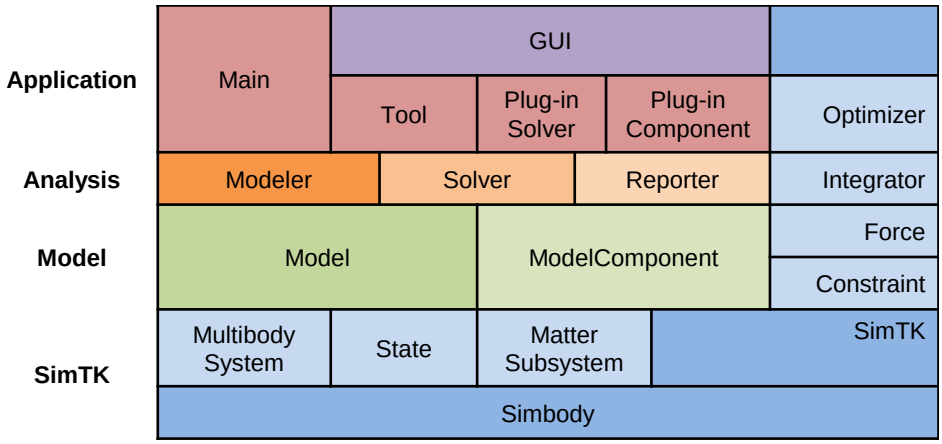



**Figure 7.1** OpenSim organization and hierarchy structure {adapted from Seth *et al.*, 2011}.

## 7.2 Development of a skeletal model of the knee joint

The OpenSim uses relative coordinates in its formulation and a skeleton is generally modeled as a set of rigid bodies interconnected by kinematic joints, which define how a child body can move with respect to its parent body. OpenSim offers seven types of kinematic joints: weld joint (also known as rigid joint), pin joint (also known as revolute joint), slider joint (also known as translational joint), ball joint (also known as spherical joint), ellipsoid joint, free joint and custom joint.

A knee joint model with a prosthetic device was built in OpenSim, which is composed by ten rigid bodies connected by nine joints. The tibiofemoral and patellofemoral articulations are considered to be free joints, being the remaining seven joints modeled as weld joints (*i.e.*, rigid joints). Thus, the knee model has twelve degrees-of-freedom that correspond to the *xyz*-translations and three rotations $\phi_1$, $\phi_2$ and $\phi_3$ of the tibiofemoral and patellofemoral joints. Figure 7.2 illustrates the knee joint model. In order to define the body-set of the model within OpenSim, some properties have to be specified, namely (*i*) body mass and its inertia properties, (*ii*) joint type and its location and orientation in relation to parent and child bodies. Additionally, the user has the option to customize some visualization features such as geometry, color,

opacity, etc (Anderson *et al.*, 2011). Within OpenSim, the geometry file used for visualization purpose has to be provided as a vtp file, which is a file type that contains polygonal data (VTK Polygonal Data) and is associated with ParaView (Kitware, Inc., Clifton Park, NY). ParaView is an open-source system developed for analyzing extremely large datasets using distributed memory computing resources.

With the purpose of developing the knee joint model shown in Figure 7.2, subject-specific data released for the "Grand Challenge Competition to Predict In Vivo Knee Loads" was utilized (Fregly *et al.*, 2012). This competition is a SimTK project and all *in vivo* data is available for download at https://simtk.org/home/kneeloads. There are two sets of *in vivo* data available in this SimTK project. The data-set labeled with the acronym JW was utilized in this work. This data-set corresponds to a male subject, age 83 years old, mass 68 kg, height 1.66 m, with a neutral leg alignment and a total knee replacement at the right knee. The interested reader in the details on the *in vivo* data acquisition and on the associated modeling approaches is referred to the works by Banks *et al.* (2005), D'Lima *et al.* (2005), Fregly *et al.* (2012), Kim *et al.* (2009).

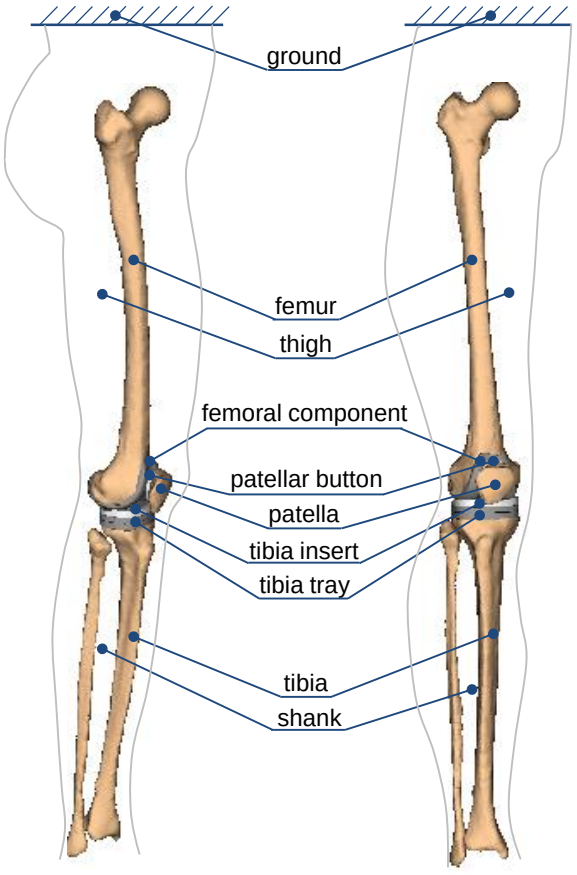


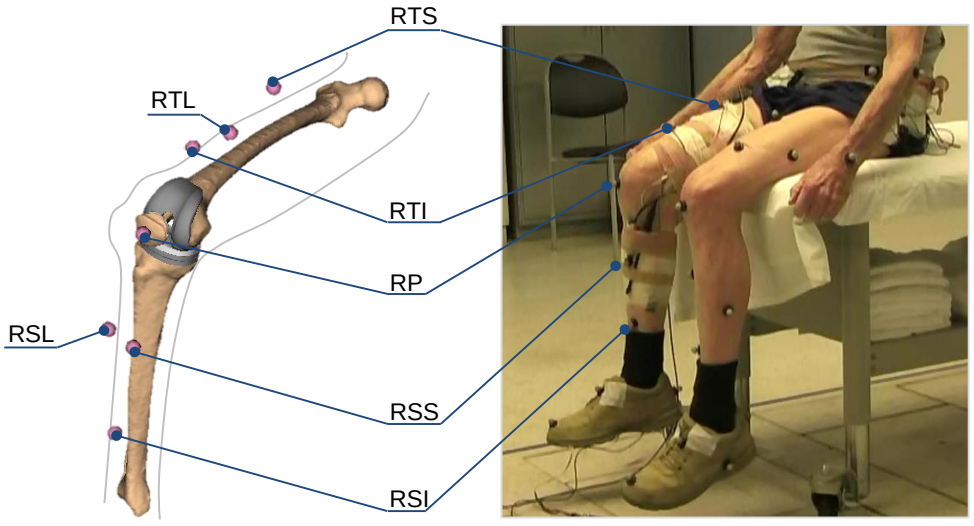

**Figure 7.2** Representation of the skeletal model of a knee joint with a prosthesis developed within OpenSim GUI: (a) lateral view; (b) anterior view.

## 7.3 Inverse kinematic analysis

After creating the knee joint model, inverse kinematics (IK) was carried out in order to find the set of generalized coordinates (joint angles and positions) for the model that best match the experimental kinematic data recorded for the JW subject. Motion data were collected for electromyography (EMG) calibration trials, static trials, isolated joint motion trials, leg motion trials and gait motion trials. For instance, the experimental data-set from the leg extension trial was utilized to perform an inverse kinematic analysis. In this trial, motion was measured using a ten-camera motion capture system (Motion Analysis Corporation, Santa Rosa, CA), being seven markers utilized. The placement of these markers is described in Table 7.1 and illustrated in Figure 7.3. Within OpenSim, a marker-set was included into the model file. The markers were placed on the body segments of the model consistently with their physical locations during the motion trial, as it can be observed in Figure 7.3.

**Table 7.1** List of markers placed at the right knee for the leg extension trial.

| Name | Acronym | Placement |
|---|---|---|
| Right Thigh Superior | RTS | Mid anterior surface of the thigh |
| Right Thigh Inferior | RTI | Distal anterior surface of the thigh |
| Right Thigh Lateral | RTL | Lateral aspect of thigh midway between RTS and RTI |
| Right Patella | RP | Center of the right patella |
| Right Shank Superior | RSS | Superior anterior bony surface of the tibia |
| Right Shank Inferior | RSI | Inferior anterior bony surface of the tibia |
| Right Shank Lateral | RSL | Lateral aspect of shank midway between RSS and RSI |




**(a)** **(b)**

**Figure 7.3** Placement of the markers on the knee: (a) OpenSim model; (b) Leg extension trial. The meaning of the marker's acronyms is listed in Table 7.1.

The IK tool in OpenSim computes the generalized coordinates that places the model in a pose compatible with the experimental marker locations in each time step. At the end of an IK task, OpenSim generated a motion file containing the generalized coordinates of the model over all time frames. Mathematically, the weighted least squares problem solved by IK tool of OpenSim is expressed as

$$\min_{\mathbf{q}} \left[ \sum_{i=1}^{\text{markers}} w_i \left\| \mathbf{x}_i^{exp} - \mathbf{x}_i(\mathbf{q}) \right\|^2 + \sum_{j=1}^{\text{unprescribed coordinates}} \omega_j (\mathbf{q}_j^{exp} - \mathbf{q}_j)^2 \right] \tag{7.1}$$

where $\mathbf{q}$ is the vector of generalized coordinates, $\mathbf{x}_i^{\text{exp}}$ is the experimental position of marker $i$, $\mathbf{x}_i(\mathbf{q})$ is the position of the corresponding marker on the model (which depends on the coordinate values), $\mathbf{q}_j^{\text{exp}}$ is the experimental value for coordinate $j$. Thus, experimental markers are matched by model markers throughout the motion by varying the joint angles through time, whose solution aims to minimize both marker and coordinate errors (Anderson *et al.*, 2011). The weighting factors of the markers and the coordinates are denoted by $w_i$ and $\omega_j$, respectively. Within OpenSim, this least squares problem is solved using a general quadratic programming solver, with a convergence criterion of $10^{-4}$ and a limit of $10^3$ iterations.

Figure 7.4 depicts the knee flexion angle along time, which is an example of a result that can be got from an inverse kinematic analysis. This motion was obtained using the experimental kinematic data recorded for the JW subject during a gait trial.

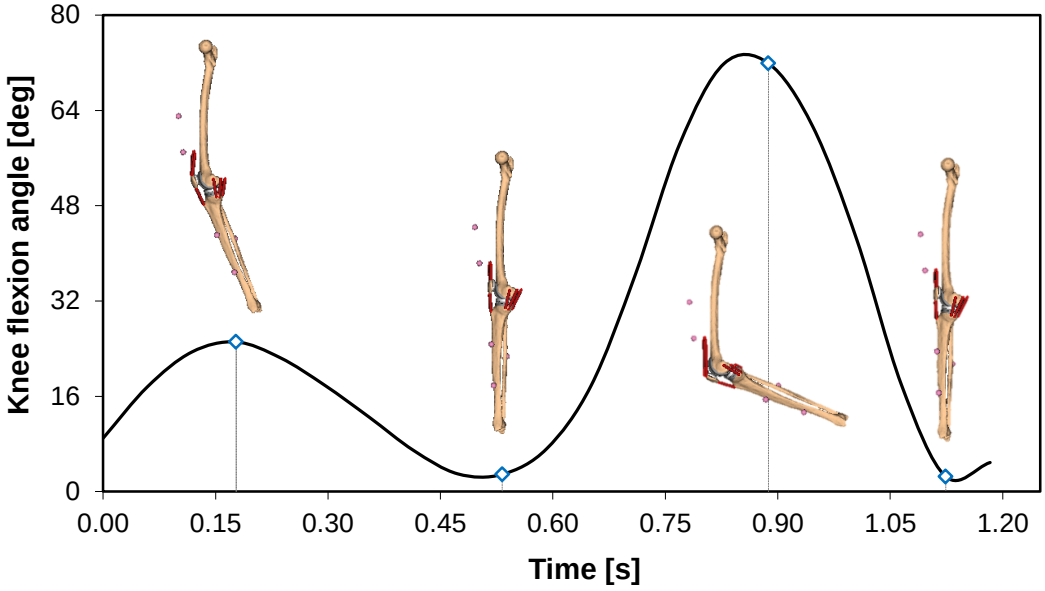



**Figure 7.4** Knee flexion angle *versus* time obtained in OpenSim by inverse kinematic analysis.

With the motion file generated by inverse kinematics, an inverse dynamics (ID) can be performed in OpenSim GUI. This type of analysis is widely used in biomechanics as it consists of a non-invasive method for calculating and evaluating the reaction forces and moments developed at the joints, as a result of performing a task that has been previously observed (Silva, 2003). In OpenSim, the ID tool determines the generalized forces at each joint responsible for a given movement.

The inverse dynamics approach is simple and efficient; however it has some limitations (Pàmies-Vilà *et al.*, 2012). For instance, the calculation of reliable accelerations may be problematic due to relatively low accuracy of the experimental measurements and the necessity of differentiating twice these measurements to obtain accelerations. The choice of filtering and smoothening methods can have a significant influence on the results (Alonso *et al.*, 2010). Additionally, errors in alignment of ground reaction force and foot may affect the calculated lower extremity forces. Moreover, only the previously measured kinematics can be analyzed (Wojtyra, 2003).

## 7.4 Muscle modeling

Fourteen musculotendinous actuators were added to the knee model to represent the main muscles and tendons responsible for the desired knee kinematics. These muscles are the sartorius, the gracilis, the biceps femoris long head, the biceps femoris short head, the gastrocnemius lateral, the gastrocnemius medial, the semimembranosus, the semitendinosus, the tensor fascia lata, the vastus medialis, the vastus intermedius, the vastus lateralis, the rectus femoris and the patellar tendon. The knee model with these fourteen musculotendinous actuators is depicted in Figure 7.5.

The musculotendinous actuators were modeled using the Thelen2003Muscle, which is a Hill-type muscle model available within OpenSim (Thelen, 2003). In order to define each musculotendinous element, a geometric path must be described, namely origin and insertion points. Within the Thelen2003Muscle model, some properties have also to be specified for each muscle, such as the maximum isometric force, $F_i^0$, the optimal fiber length, $l_i^0$, the tendon slack length, $l_t^0$, and the angle between tendon and fibers at optimal fiber length (pennation angle), $\alpha_i$, etc.

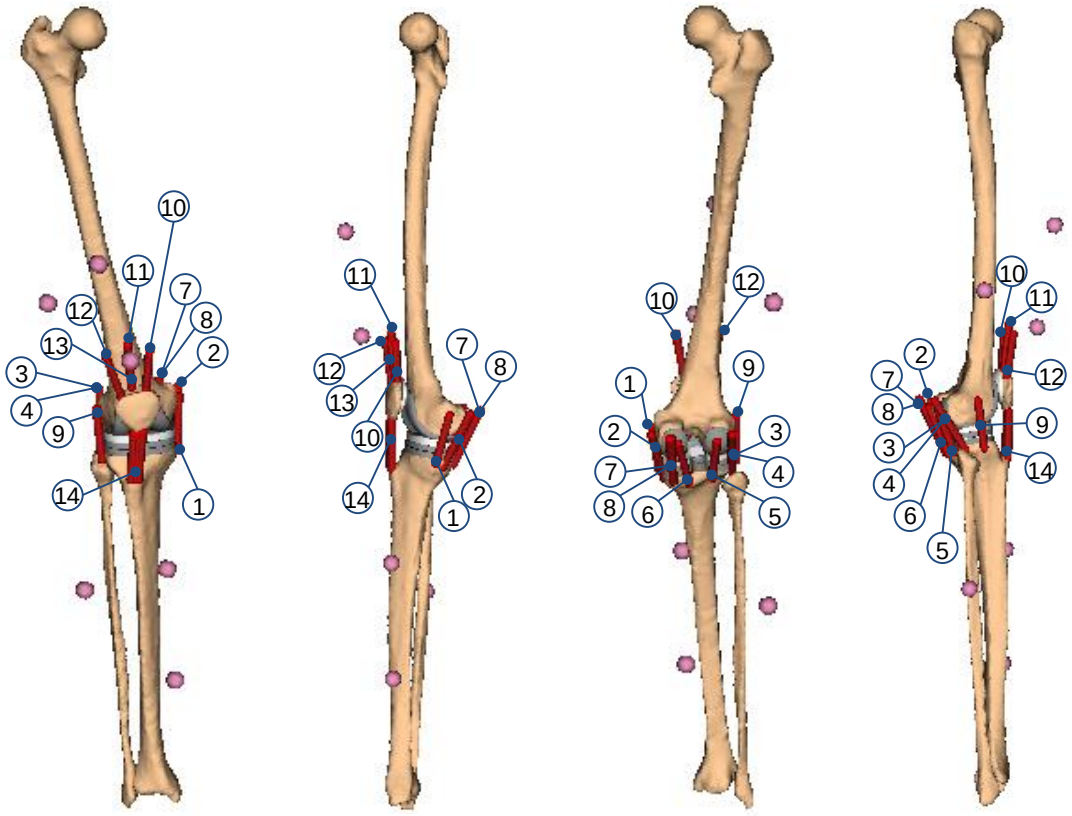



**Figure 7.5** Schematic representation of the knee model with fourteen muscle actuators: (a) anterior view; (b) medial view; (c) posterior view; (d) lateral view. The fourteen muscle actuators represented in this figure are numbered from 1 to 14, representing: (1) Sartorius, (2) Gracilis, (3) Biceps femoris long head, (4) Biceps femoris short head, (5) Gastrocnemius lateral, (6) Gastrocnemius medial, (7) Semimembranosus, (8) Semitendinosus, (9) Tensor fascia lata, (10) Vastus medialis, (11) Vastus intermedius, (12) Vastus lateralis, (13) Rectus femoris, (14) Patellar tendon.

OpenSim offers two different methods for estimating muscle activations: static optimization (SO) and computed muscle control (CMC). SO is an extension to inverse dynamics that resolves the net joint moments into individual muscle forces at each instant in time (Ackermann, 2007; Anderson *et al.*, 2011). The SO tool uses the known motion of the model to solve the equations of motion for the unknown generalized forces subject to one of the following muscle activation-to-force conditions

$$\underbrace{\sum_{i=1}^{n_i}\left(a_i F_i^0\right) r_{i,k} = \tau_k}_{\text{ideal force generators}} \tag{7.2}$$

or

$$\underbrace{\sum_{i=1}^{n_i}\left[a_i f\left(F_i^0, l_i, v_i\right)\right] r_{i,k} = \tau_k}_{\text{constrained by force-length-velocity properties}} \tag{7.3}$$

while minimizing the objective function,

$$J = \sum_{i=1}^{n_i}\left(a_i\right)^p \tag{7.4}$$

where $n_i$ is the number of muscles included into the model, $a_i$ represents the activation level of muscle $i$ at a discrete time step, $l_i$ denotes the muscle fiber length, $v_i$ is the muscle shortening velocity, $r_{i,k}$ represents the muscle moment arm about the $k$-th joint axis, $\tau_k$ denotes the generalized force acting about the $k$-th joint axis, $f\left(F_i^0, l_i, v_i\right)$ is force-length-velocity surface that represents the contractile tissue behavior and $p$ denotes a user-defined constant that is related to individual muscle properties such as fiber type, fiber orientation, muscle fatigue, etc (Ackermann, 2007; Ou, 2011). According to Ou (2011), experimental studies demonstrated that the $p$ power has a range from 1.4 to 5.1. It is worth noting that Equation (7.3) expresses the muscle activation-to-force relation in a more physiologic way than Equation (7.2), since Equation (7.3) takes into account muscle properties, such as the force-length relation and the force-velocity relation that are depicted in Figures 7.6a and 7.6b, respectively.

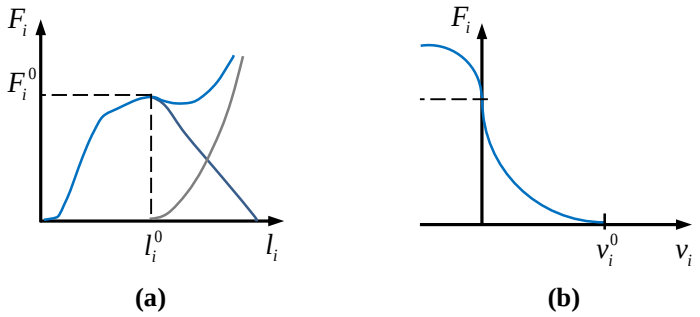



**Figure 7.6** Muscle properties: (a) force-length relation; (b) force-velocity relation.

Computed muscle control computes a set of muscle excitation levels that drive the generalized coordinates of the dynamic model towards a desired kinematic trajectory. The CMC tool combines a proportional-derivative (PD) control, a static optimization and a forward dynamic simulation. Figure 7.7 depicts how CMC analysis is carried out step-by-step. Before starting the CMC algorithm, initial conditions of the model are computed, which comprise the generalized coordinates, generalized velocities, plus any muscle conditions (i.e., muscle activation levels and muscle fiber lengths). While the

initial values of generalized coordinates and velocities can be obtained from the desired kinematics, the initial muscle conditions are generally unknown. In order to compute appropriate initial muscle conditions, CMC is applied to the first 0.03 seconds of the desired movement. Because the muscle conditions are generally out of equilibrium and muscle forces can change significantly during this initial time interval, the output results of this interval of simulation are generally not valid. Therefore, it is important to guarantee that the CMC starts at least 0.03 seconds prior to the desired interval (Anderson *et al.*, 2011).

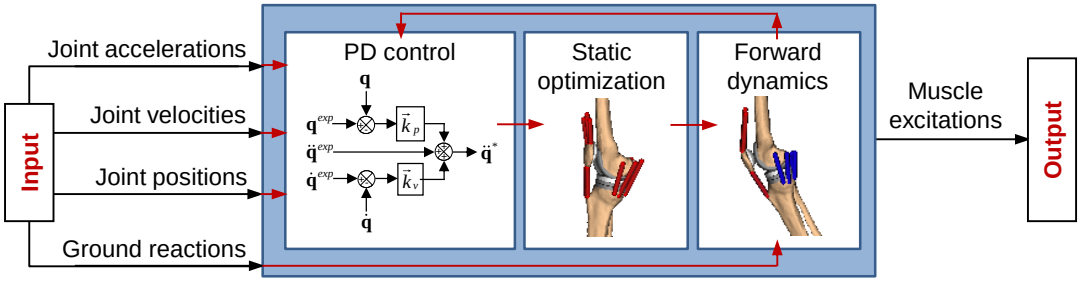



**Figure 7.7** Schematic representation of the CMC algorithm.

The first step of the CMC algorithm is to compute a set of desired accelerations, $\ddot{\mathbf{q}}^*$, which when achieved will drive the model coordinates, $\mathbf{q}$, toward the experimentally-derived coordinates, $\mathbf{q}^{exp}$. The desired accelerations are computed using a proportional-derivative control law that can be written as (Anderson *et al.*, 2011)

$$\ddot{\mathbf{q}}^*\left(t+\Delta t\right)=\ddot{\mathbf{q}}^{exp}\left(t+\Delta t\right)+k_v\left[\dot{\mathbf{q}}^{exp}\left(t\right)-\dot{\mathbf{q}}\left(t\right)\right]+k_p\left[\mathbf{q}^{exp}\left(t\right)-\mathbf{q}\left(t\right)\right] \tag{7.5}$$

where $k_v$ and $k_p$ are the feedback gains on the velocity and position errors, respectively.

The error gains of $k_v$=20 and $k_p$=100 cut down tracking errors, being, therefore, indicated for musculoskeletal simulations within CMC tool. The next step in CMC is to compute the actuator controls that will achieve the desired accelerations. Any type of actuator can be used. In this step, static optimization is applied to distribute the loads across the synergistic actuators. The final task of CMC is to use the computed controls to carry out a forward dynamic simulation. These steps, computing the desired accelerations, static optimization and forward dynamics, are repeated until time is advanced to the end of the desired movement interval, as Figure 7.7 shows (Neptune and Kautz, 2001; Thelen *et al.*, 2003; Anderson *et al.*, 2011).

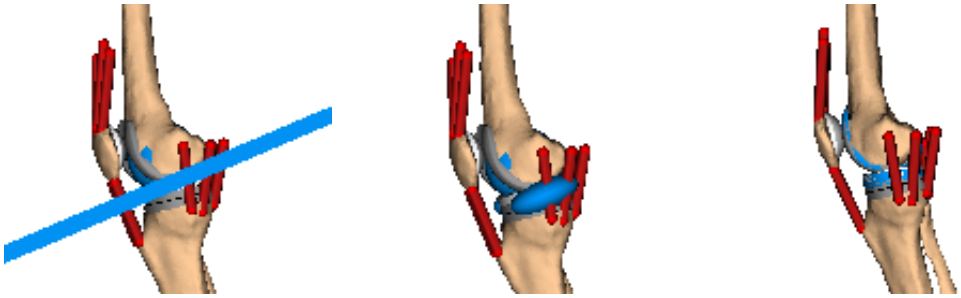
## 7.5 Contact forces modeling and analysis

In OpenSim, the contact analysis involves the geometric description of the contact surfaces and the application of an appropriate constitutive law to compute the contact forces. Regarding the contact geometries, three types of surface models are currently available in OpenSim, namely planes, spheres and triangular meshes. When the contacting bodies present simple geometries, planes and spheres are used to describe the contact surfaces. In turn, when complex geometries are involved in the contact event, more accurate geometries are demanded and, hence, triangular mesh volumes are utilized. It is worth mentioning that the three contact surface models can be applied in the same system. In order to define a contact geometry within OpenSim it is necessary to specify the contacting body and the location and orientation of the contact surface on the body reference frame. Moreover, if the geometry is spherical, a radius must be indicated, while for a triangular mesh an .obj file is required. This .obj file has to represent a closed manifold and must be generated by the MeshLab (an open-source computational tool developed with the support of the 3D-CoForm project, available from: http://meshlab.sourceforge.net/).

OpenSim provides two compliant contact formulations. One is based on the Hunt and Crossley model, which analytically computes forces based on assumptions of the Hertzian theory of elasticity. Other is the Elastic foundation model that calculates contact forces using a simplified bed-of-springs model. Both approaches are augmented with a dissipation term and a Stribeck model in order to account for viscous damping and friction effects, respectively (Seth *et al.*, 2011). An option to model contact interactions via a combination of non-penetrating unilateral constraints is also available in OpenSim. This contact formulation is out of the scope of this work. The interested reader on contact modeling within OpenSim using unilateral constraints is referred to the work by Hamner *et al.* (2010). It is worth noting that the selection of a constitutive law to evaluate contact forces is not an arbitrary choice as it depends on the model used to define the geometry of the contact surfaces. For instance, Hunt and Crossley law must be applied when the surfaces are modeled as spheres or half-spaces, while the Elastic foundation model is appropriate for cases where at least one of the surfaces is defined as a triangular mesh volume.

### 7.5.1 Prediction of knee joint contact forces by inverse dynamics

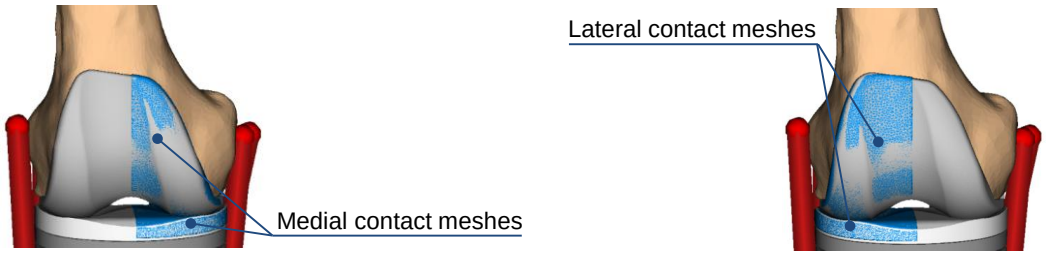
Within this study, three geometric models were considered to define the knee articular surfaces. Figure 7.8a shows the first model that corresponds to a simple sphere-plane configuration. In the second model, the planar geometry of the tibial tray was replaced by two ellipsoids that were modeled using triangular mesh volumes, as it can be observed in Figure 7.8b. This second model has two contacting pairs that permits to evaluate the medial contact forces and the lateral ones. Finally, the triangular mesh volumes of the original CAD files of the femoral component and tibial tray were utilized to describe the contact geometries. This third model is depicted in Figure 7.8c.



**(a)** **(b)** **(c)**

**Figure 7.8** Geometrical models used to model knee contact surfaces: (a) sphere-plane contact; (b) two sphere-ellipsoid contacts (medial and lateral); (c) contact between meshes of CAD files.

Despite considering CAD-based geometries, the knee model illustrated in Figure 7.8c only accounts for a single contact pair between the femoral and tibial components. In order to distinguish the medial contact forces from the lateral ones, new geometries were included into the model. These new geometries were generated by splitting both CAD models into two parts: the medial side and the lateral side, as Figure 7.9 shows.




**(a)** **(b)**

**Figure 7.9** Triangular mesh volumes generated by CAD files of knee prosthesis: (a) Medial contact meshes, (b) Lateral contact meshes.

With the intent to study the influence of articular geometry on the contact response of the knee model, dynamic simulations using the three models of Figure 7.8 were carried out within OpenSim GUI. The motion of the knee joint during a gait trial

was utilized for this study. The variation of the knee flexion angle along time of this motion is shown in Figure 7.4. For this analysis, the joint motions generated by a previously inverse kinematics are used to run inverse dynamics and to estimate the tibiofemoral contact forces. The results are plotted in Figures 7.10, 7.11 and 7.12.

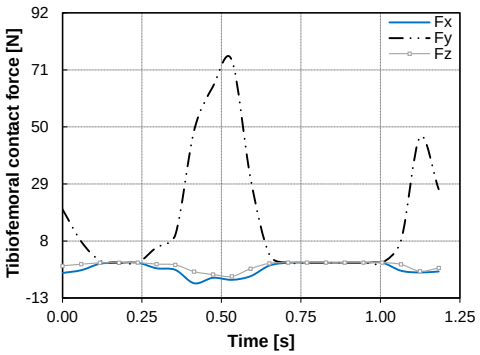
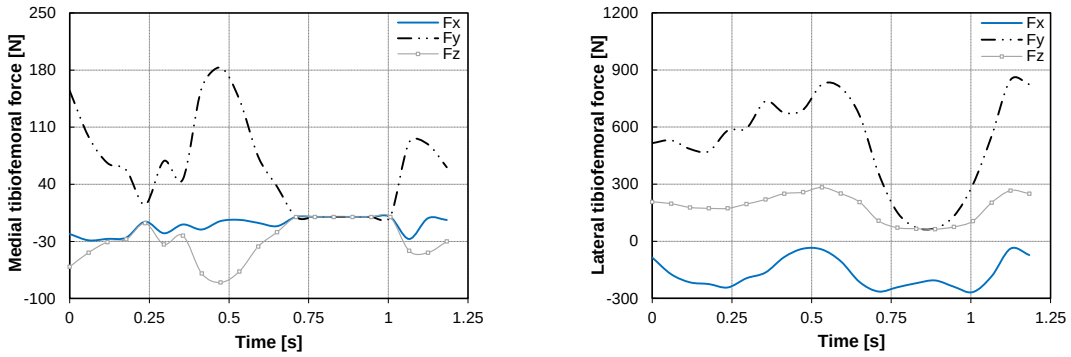
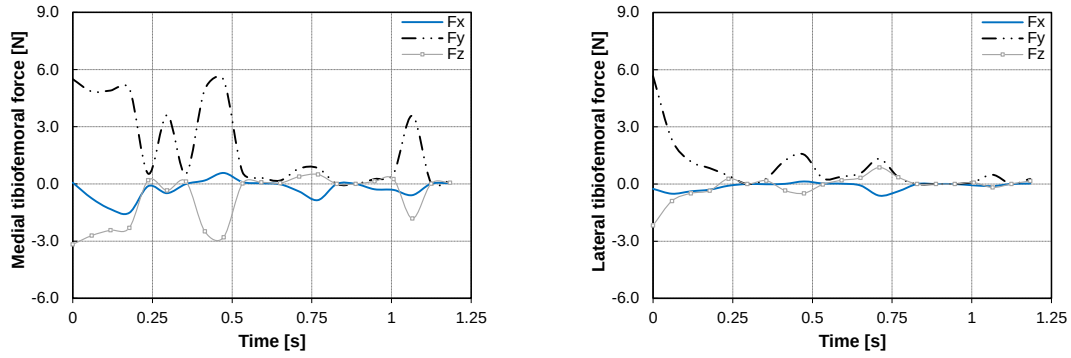



**Figure 7.10** Components of the tibiofemoral contact force on *x*-, *y*- and *z*- directions for the sphere-plane geometric model




**(a)** **(b)**

**Figure 7.11** Components of the tibiofemoral contact force on *x*-, *y*- and *z*- directions for ellipsoid-sphere geometric model: (a) medial forces; (b) lateral forces.




**(a)** **(b)**

**Figure 7.12** Components of the tibiofemoral contact forces on *x*-, *y*- and *z*- directions for CAD-based models: (a) medial forces; (b) lateral forces.

The solution of a contact problem for multibody dynamics requires modeling and analyzing the contact process that depends on many factors, such as the geometry of the

contacting surfaces, the material properties of the contacting bodies and the constitutive law considered to represent the interaction among the different bodies that comprises the multibody system. In general, formulations for contact modeling and analysis comprise two main steps, namely: (*i*) the geometrical detection of contact and (*ii*) the evaluation of the contact forces, which are the result of collisions between bodies

By analyzing Figures 7.10-12, it can be concluded that the geometrical model has a great influence on the knee contact forces, namely in the force patterns and magnitudes. Nonetheless, this effect can be also associated with other issues of the contact formulation utilized by OpenSim to solve each problem, namely the contact detection approach or the contact force model. Therefore, further analyses must be performed to understand the differences on the results plotted in Figures 7.10-12.

With the purpose of investigating the influence of the contact force model on the dynamic results of the knee joint, two additional simulations were performed using the sphere-plane knee model illustrated in Figure 7.8a. The Elastic foundation model (EFM) was utilized in the first simulation, being the Hunt and Crossley model (HCM) applied in the second test. As mentioned previously, the use of the EFM within OpenSim to compute contact forces demands the description of at least one of the geometries as a contact mesh. Thus, in the first simulation, the contact sphere was defined by means of triangular mesh volume (discrete geometrical representation); while in the second test it was modeled via an algebraic function (continuous geometrical representation). The obtained results are plotted in Figure 7.13, which depicts some differences between the force histories. The contact forces computed by Hunt and Crossley law are higher than those calculated when the Elastic foundation model is applied. This outcome was not expected since the same material properties were considered in both models. Though, the results can be justified by the differences on the geometrical models applied to describe the contact sphere. It is believed that discrete and continuous geometric models require distinct approaches for contact detection, which may lead to differences on the computation of the relative indentation and, hence, on the contact forces.

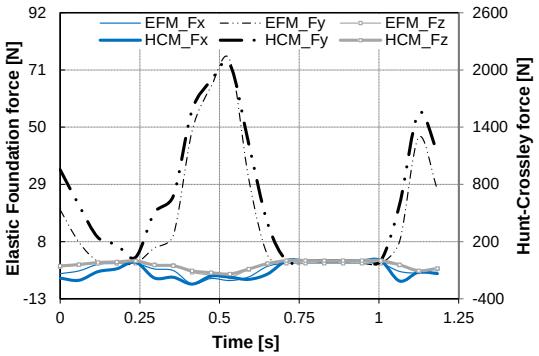



**Figure 7.13** Components of the tibiofemoral contact force on *x*-, *y*- and *z*- directions for the plane-sphere geometric model computed by the use of the Elastic foundation model (EFM) and the Hunt and Crossley model (HCM).

### 7.5.2 Study on the accuracy of OpenSim contact models

In order to evaluate the accuracy of OpenSim contact formulations for predicting the dynamic response of multibody systems with contact, additional dynamic analyses were carried out using a bouncing ball system. This classic problem of contact mechanics was considered for validation purposes because it is one of the simplest contact systems and it has been widely used to corroborate methods and formulations associated with contact (Flores *et al.*, 2011).

Within OpenSim, two forward dynamic simulations were performed using a bouncing ball system. In the first simulation, the Hunt and Crossley model was utilized to evaluate the contact forces, while in the second simulation the contact forces were computed using the Elastic foundation model. A computational simulation using the bouncing ball model was also carried out in MUBODYNA code (Flores, 2010). In MUBODYNA, the Hertz contact law was utilized to compute the normal contact forces. The ball contacts were modeled as purely elastic. Therefore, the dissipation parameter of the Hunt and Crossley model and the Elastic foundation model was set to zero in OpenSim. For sake of comparison and discussion purposes, the bouncing ball problem was also solved analytically.

Figure 7.14 shows an elastic bouncing ball with an initial height equal to 0.4 m, a mass of 1.0 kg, a radius of 0.3 m and a relative contact stiffness parameter equal to $2.582\times10^{7}$ N/m$^{1.5}$. The ball is acted upon by gravitational force only which is taken as acting in the negative *y*-direction, being the ground body considered rigid and stationary. It must be mentioned that a bouncing ball system similar to the model presented in Figure 7.14 is available into the SimtK project entitled “Contact Modeling with OpenSim: a User’s Approach” developed by Lopes and Machado in March 2012.

This project aimed to play as a supplement for OpenSim documentation on contact modeling. It comprises two step-by-step tutorials and it can be downloaded from the website https://simtk.org/home/contact_osim_u.

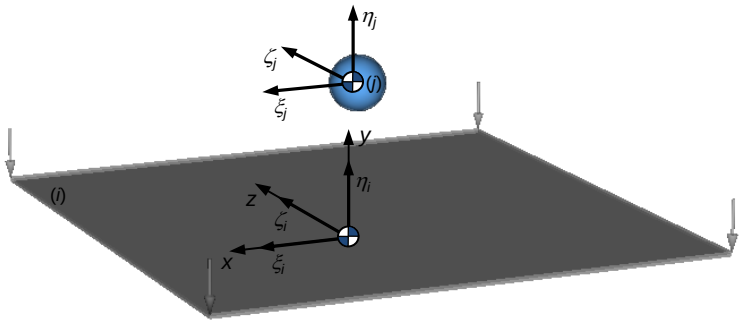



**Figure 7.14** Bouncing ball model used in OpenSim simulations.

When the ball contacts with the ground, the contact indentation is expressed as

$$\delta = y_b - R \tag{7.6}$$

where $y_b$ is the *y*-coordinate of the ball center of mass and $R$ represents the ball radius. The instant of time when a ball hits the ground for the first time ($t^-$) is given by

$$t^- = \sqrt{\frac{2\left(y_0 - R\right)}{a_y}} \tag{7.7}$$

in which $y_0$ is the initial height of the ball and $a_y$ denotes the gravitational acceleration. At the initial instant of simulation, the ball energy is expressed as

$$T^0 = ma_y\left(y_0 - R\right) \tag{7.8}$$

where $m$ is the ball mass, having the remaining variables the meaning described above. In turn, the ball energy at the instant of maximum indentation can be written as

$$T^- = T^{max} + U^{max} + \Delta E \tag{7.9}$$

where $T^{max}$ represents the kinetic energy of the system at the end of the compression phase, $U^{max}$ is the maximum elastic strain energy stored and $\Delta E$ denotes the dissipated energy associated with internal damping of the material. The kinetic energy of the system at the end of the compression phase is zero because the ball has null velocity. Within the bouncing ball example, the collision between the ball and the ground is considered purely elastic, which means that none energy is dissipated during the contact

($\Delta E_c$=0). The stored strain energy is equal to the work done by the contact force that develops from the state of zero deformation to the state of maximum deformation. Considering the Hertz contact law, the maximum elastic strain energy stored is given by

$$U^{max} = \frac{2}{5} K \delta_{max}^{\frac{5}{2}} \tag{7.10}$$

where $\delta_{max}$ represents the maximum indentation. Thus, the balance of energy between the initial instant of simulation ($t_0$) and the instant of maximum contact indentation, $t_{max}$, can be written as

$$ma_y \left(y_0 - R\right) = \frac{2}{5} K \delta_{max}^{\frac{5}{2}} \tag{7.11}$$

Based on Equations (7.6)-(7.11), the values of some variables were solved analytically and listed in Table 7.2, namely the ball energy at the initial instant of simulation, $T^0$, the first instant of contact for the first impact, $t^-$, and the relative indentation of the first instant of contact for the first impact, $\delta^-$. The results obtained from the computational simulations performed in OpenSim and MUBODYNA are plotted Figures 7.15-17 and listed in Tables 7.3-7.4.

**Table 7.2** Analytical solution of the bouncing ball problem. This table includes the values of following variables: the ball energy at the initial instant of simulation ( $T^0$ ), the first instant of contact for the first impact ( $t^-$ ) and the relative indentation of the first instant of contact for the first impact ( $\delta^-$ ).

| $T^0$ [J] | $t^-$ [s] | $\delta^-$ [mm] |
|---|---|---|
| 0.98100 | 0.14279 | 0.00797 |

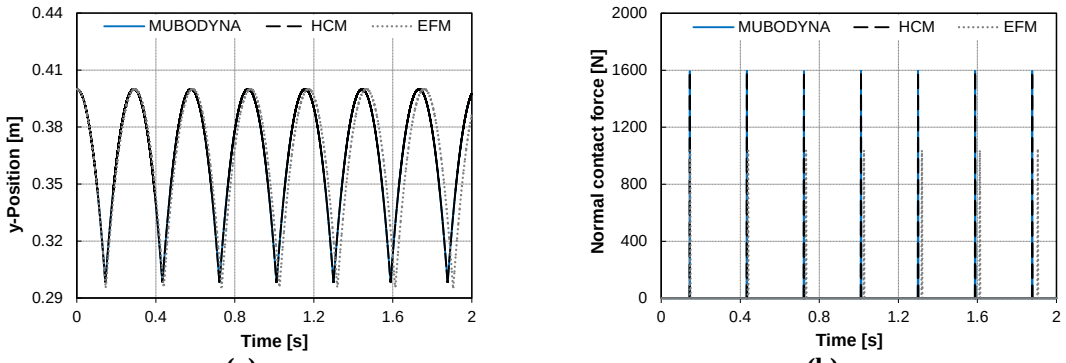
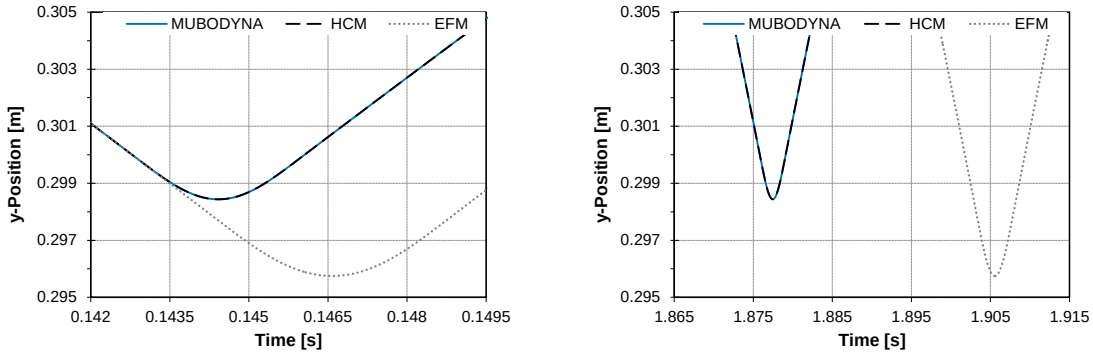



**Figure 7.15** Ball contact response using different contact force laws, namely Hertz contact law within MUBODYNA, Hunt and Crossley model (HCM) within OpenSim and Elastic foundation model (EFM) within OpenSim: (a) *y*-position of the ball *versus* time; (b) normal contact forces *versus* time.

**Table 7.3** Ball contact response obtained by using different contact models, namely Hertz law within MUBODYNA, Hunt and Crossley model within OpenSim and Elastic foundation model within OpenSim. This table includes the values of following variables: the first instant of contact for the first impact ($t^-$); the relative indentation of the first contact of the first impact ($\delta^-$); the normal contact force of the first contact of the first impact ($F^-$); the instant of maximum indentation for the first impact ($t_{max}$); the maximum indentation for the first impact ($\delta_{max}$); the maximum contact force for the first impact ($F_{max}$); the ball energy at instant of maximum indentation for the first impact ($U^{max}$).

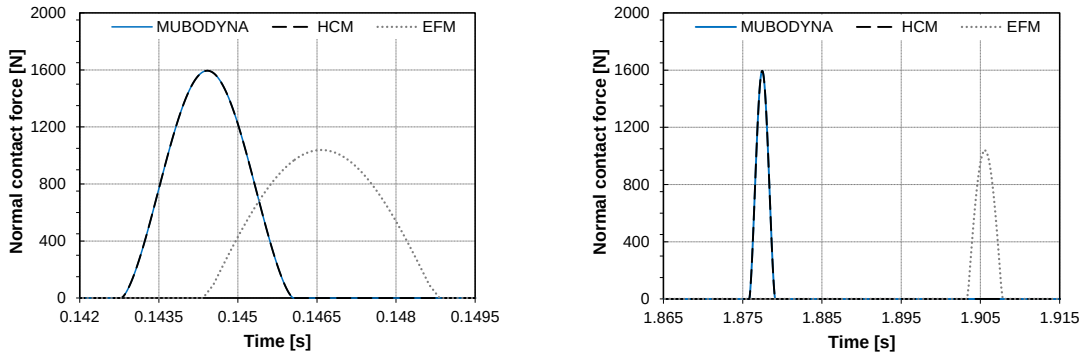
| | Hertz law | Hunt and Crossley model | Elastic foundation model |
|---|---|---|---|
| $t^-$ [s] | 0.14279 | 0.14279 | 0.14434 |
| $\delta^-$ [mm] | 0.00797 | 0.00797 | 2.18921 |
| $F^-$ [s] | 0.58063 | 0.58063 | 0.62180 |
| $t_{max}$ [s] | 0.14442 | 0.14442 | 0.14658 |
| $\delta_{max}$ [mm] | 1.56230 | 1.56230 | 4.25117 |
| $F_{max}$ [$N$] | 1594.36 | 1594.36 | 1038.27 |
| $U^{max}$ [J] | 0.99600 | 0.99600 | 12.1650 |

From Figure 7.15, it can be observed that the ball starts the simulation at an initial position of 0.4 m high from the ground and falls down until it impacts with the ground for the first time. When the ball collides with the ground, a contact takes place and the ball rebounds, producing jumps. The ball hits the ground seven times during the 2 s of simulation. After each impact, the ball returns to its initial position, which means that no energy dissipation is accounted during the impact process. The physical behavior of a purely elastic bouncing ball was reproduced by the three contact force approaches, as in all simulations the ball only undergoes vertical motion and the value of the maximal contact force is the same in the seven impacts.




**(a)** **(b)**

**Figure 7.16** Vertical position of the ball using different contact force laws, namely Hertz contact law within MUBODYNA, Hunt and Crossley model (HCM) within OpenSim and Elastic foundation model (EFM) within OpenSim: (a) First impact; (b) Last impact (7$^{th}$ impact).




**(a)** **(b)**

**Figure 7.17** Normal contact forces of the ball using different contact laws, namely Hertz contact law within MUBODYNA, Hunt and Crossley model (HCM) within OpenSim and Elastic foundation model (EFM) within OpenSim: (a) First impact; (b) Last impact ($7^{th}$ impact).

Figures 7.16a and 7.17a depict the vertical position and the normal contact forces of the bouncing ball for the first impact, while Figures 7.16b and 7.17b are associated with the last impact ($7^{th}$ impact). The results obtained by using Hertz law within MUBODYNA are equal to the ones reported by OpenSim when the Hunt and Crossley model is considered, as it can be observed in Figures 7.15-7.17 and Tables 7.3 and 7.4. This outcome was expected because both contact force models utilized continuous representations to describe the contact geometries. According to the principle of energy conservation, the ball energy at the initial instant of time, $T^0 = 0.981$ J, is equal to the ball energy at the instant of maximum deformation, $U^{max} = 0.996$ J. Nevertheless, a difference of 0.015 J is reported between the analytical solution and the computational results of the simulations using Hertz law and Hunt and Crossley model. This small difference is due to numerical errors of integration, common to both computational tools OpenSim and MUBODYNA, that are related to the type of the integration method and the associated time parameters, such as time-step and time of contact detection.

The contact response of the ball resultant from the OpenSim simulation that uses the Elastic foundation model is different from the other two sets of results and the analytical solution. In Figure 7.16, it is visible that the OpenSim elastic foundation model allows higher contact indentations before applying the rebound contact force. Figure 7.15b shows that the Elastic foundation model computes lower contact forces than the Hunt and Crossley model. At the knee joint application case, the Hunt and Crossley model reports also higher contact forces than the Elastic foundation model (see Figure 7.13). The higher indentations and the lower contact forces predicted by the Elastic foundation model lead to significant variations on the dynamics of the system. The first instants of contact of the seven impacts of the ball using different contact

models are listed in Table 7.4. In the simulation using the Elastic foundation model, the ball hits the ground for the first time 1.55 ms after the other simulations. Moreover, the impact duration is larger when the Elastic foundation model is applied. These two differences on the first impact affect the dynamic of the system and lead to more accentuated discrepancies through time, being the delay on the first instant of contact of the last impact (the 7th impact) equal to 27.49 ms. These outcomes suggested that the formulations adopted by OpenSim to detect contact events between discrete geometrical representations (triangular mesh volumes) requires further investigation, improvement and validation (Machado *et al.*, 2012). The verification of the implementation process of the Elastic foundation model in the OpenSim software is also recommended.

**Table 7.4** First instant of contact ($t^-$) of the seven impacts of the bouncing ball and impact duration ($\Delta t_{impact}$) using different contact models, namely Hertz law within MUBODYNA, Hunt and Crossley model within OpenSim and Elastic foundation model within OpenSim.

| | Hertz law | Hunt and Crossley model | Elastic foundation model |
|---|---|---|---|
| | $t^-$ [s] | $t^-$ [s] | $t^-$ [s] |
| 1st impact | 0.14279 | 0.14279 | 0.14434 |
| 2nd impact | 0.43163 | 0.43163 | 0.43751 |
| 3rd impact | 0.72047 | 0.72047 | 0.73067 |
| 4th impact | 1.00931 | 1.00931 | 1.02384 |
| 5th impact | 1.29815 | 1.29815 | 1.31699 |
| 6th impact | 1.58699 | 1.58699 | 1.61016 |
| 7th impact | 1.87583 | 1.87583 | 1.90332 |
| $\Delta t_{impact}$ [s] | 0.00326 | 0.00326 | 0.00448 |

## 7.6 Summary and discussion

In this Chapter, the capabilities of OpenSim on biomechanical modeling and analysis were explored. Thus, a four-step framework for contact and muscle modeling within OpenSim was offered. The first task consists of developing a skeletal model. OpenSim was developed using multibody system methodologies and, therefore, the anatomical segments are described as rigid bodies that are kinematically constrained by joints. In this work, a knee joint model with prosthesis was presented, which is composed by ten rigid bodies connected by nine joints.

Secondly, an inverse kinematics was carried out in OpenSim. For this analysis, a leg extension motion was considered, which was acquired by means of a motion capture system. The markers were placed on the model consistently with their physical locations

in the motion trial. The inverse kinematics tool in OpenSim computes the generalized coordinates that places the model in a pose compatible with the experimental marker locations in each time step. At the end of this analysis, OpenSim generates a motion file that contains the generalized coordinates of the model segments over all time frames.

The third step is to introduce into the model the main muscles and associated tendons responsible for the desired kinematics of the model. Fourteen musculotendinous actuators were added to the knee model. To define each musculotendinous element, a geometric path was specified, as well as some muscle properties. OpenSim offered two methods for estimating muscle activations: static optimization and computed muscle control. The differences of these two formulations were pointed out in this Chapter.

The final task consists of defining a geometrical representation for the contact surfaces and applying an appropriate constitutive law to compute the contact forces. OpenSim supports three types of surface models for contact: planes, spheres and triangular mesh volumes. Regarding the constitutive law, OpenSim provides two compliant contact approaches: Hunt and Crossley model and Elastic foundation model.

In order to assess the accuracy of OpenSim formulations for predicting contact forces, several computational studies were performed. Firstly, three geometrical models were considered to define the articular surfaces, being the contact forces evaluated by using Elastic foundation model. The outcomes revealed that the contact geometry has a great influence on the knee contact forces. In the second study, Hunt and Crossley model and Elastic foundation model were utilized to compute the contact forces within the same knee model, which presents a sphere-plane contact configuration. In this set of simulations, the Hunt and Crossley model reports higher contact forces than the Elastic foundation model. In order to get an explanation for these results, a study on the accuracy of the OpenSim contact formulations was performed using a bouncing ball system. Within OpenSim, the contact response of the bouncing ball system was analyzed using the Hunt and Crossley model and using the Elastic foundation model. A computational simulation using the bouncing ball model was also performed in MUBODYNA, being the Hertz contact law utilized to compute the normal contact forces. For sake of comparison and discussion, the bouncing ball problem was also solved analytically. The computational results demonstrated that when the Hertz law within MUBODYNA or the Hunt and Crossley model within OpenSim are applied, the contact response of the ball is equal to the analytical solution. Nevertheless, the contact

response of the bouncing ball resultant from the OpenSim simulation that uses the Elastic foundation model as contact force approach is different from the other two sets of computational results. When the Elastic foundation model is utilized, the first impact is detected later and has a larger duration, which affects the dynamic of the system and leads to more accentuated differences through time.

In a broad sense, OpenSim is a useful tool for modeling, visualizing and analyzing human movements. Regarding contact analysis, this software is able to predict with reasonable accuracy contact forces between regular shapes within simple multibody systems. For the cases where volumetric meshes are used to represent the contact surfaces, OpenSim contact formulation requires further investigation, improvement and validation. In addition, combining dynamic musculoskeletal models with articular contact models is still a challenging issue demanding extra research effort.

## References


Ackermann, M. (2007) *Dynamics and energetics of walking with prostheses*. PhD Thesis. University of Stuttgart, Stuttgart, Germany.

Agarwal, P., Narayanan, M.S., Lee, L.-F., Mendel, F., Krovi, V.N. (2010) Simulation-based design of exoskeletons using musculoskeletal analysis. *Proceedings of the ASME 2010 International Design Engineering Technical Conferences* (8p.), Montreal, Canada.

Alonso, F.J., Cuadrado, J., Lugrís, U., Pintado, P. (2010) A compact smoothing-differentiation and projection approach for the kinematic data consistency of biomechanical systems. *Multibody System Dynamics*, 24(1), pp. 67-80.

Anderson, F., Guendelman, E., Habib, A., Hammer, S., Holzbaur, K., John, C., Ku, J., Liu, M., Loan, P., Reinbolt, J., Seth, A., Delp, S. (2011) *OpenSim User's Guide, Release 2.4*.

Banks, S.A., Fregly, B.J., Boniforti, F., Reinschmidt, C., Romagnoli, S. (2005) Comparing in vivo kinematics of unicondylar and bi-condylar knee replacements. *Knee Surgery, Sports Traumatology, Arthroscopy*, 13(7), pp. 551-556.

Dao, T.T., Marin, F., Pouletaut, P., Charleux, F., Aufaure, P., Ho Ba Tho, M.C. (2012) Estimation of accuracy of patient-specific musucloskeletal modelling: Case study on a post polio residual paralysis subject. *Computer Methods in Biomechanics and Biomedical Engineering*, 15(7), pp. 745-751.

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

Delp, S.L., Loan, J.P., Hoy, M.G., Zajac, F.E., Topp, E.L., Rosen, J.M. (1990) An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *IEEE Transactions on Biomedical Engineering*, 37(8), pp. 757-767.

D'Lima, D.D., Townsend, C.P., Arms, S.W., Morris, B.A., Colwell Jr, C.W. (2005) An implantable telemetry device to measure intra-articular tibial forces. *Journal of Biomechanics*, 38(2), pp. 299-304.

Flores, P. (2010) *MUBODYNA - A FORTRAN program for dynamic analysis of planar multibody systems*. University of Minho, Guimarães, Portugal.

Flores, P., Machado, M., Silva, M.T., Martins, J.M. (2011) On the continuous contact force models for soft materials in multibody dynamics. *Multibody System Dynamics*, 25(3), pp. 357-375.

Fregly, B.J., Besier, T.F., Lloyd, D.G., Delp, S.L., Banks, S.A., Pandy, M.G., D'Lima, D.D. (2012) Grand challenge competition to predict in vivo knee loads. *Journal of Orthopaedic Research*, 30(4), pp. 503-513.

Hamner, S.R., Seth, A., Delp, S. L. (2010) Muscle contributions to propulsion and support during running. *Journal of Biomechanics*, 43(14), pp. 2709-2716.

Kim, H.J., Fernandez, J.W., Akbarshahi, M., Walter, J.P., Fregly, B.J., Pandy, M.G. (2009) Evaluation of predicted knee-joint muscle force during gait using an instrumented knee implant. *Journal of Orthopaedic Research*, 27(10), pp. 1326-1331.

Lin, Y.-C., Walter, J.P., Banks, S.A., Pandy, M.G., Fregly, B.J. (2010) Simultaneous prediction of muscle and contact forces in the knee during gait. *Journal of Biomechanics*, 43(5), pp. 945-952.

Machado, M., Flores, P., Walter, J.W., Fregly, B.J. (2012) Challenges in using OpenSim as a multibody design tool to model, simulate, and analyze prosthetic devices: a knee joint case-study. *EUROMECH Colloquium 524 - Multibody system modeling, control and simulation for engineering design*. Enschede, Netherlands, 28-29.

Mihalko, W., Conner, D.J., Benner, R., Williams, J.L. (2012) How does TKA kinematics vary with transverse plane alignment changes in a contemporary implant? *Clinical Orthopaedics and Related Research*, 470(1), pp. 186-192.

Neptune, R.R., Kautz, S.A. (2001) Muscle activation and deactivation dynamics: the governing properties in fast cyclical human movement performance. *Exercise and Sport Sciences Reviews*, 29(2), pp. 76-81.

Ou, Y. (2011) *An Analysis of Optimization Methods for Identifying Muscle Forces in Human Gait.* PhD Thesis, University of Duisburg-Essen, Duisburg, Germany.

Pàmies-Vilà, R., Font-Llagunes, J.M., Cuadrado, J., Alonso, F.J. (2012) Analysis of different uncertainties in the inverse dynamic analysis of human gait. *Mechanism and Machine Theory*, 58, pp. 153-164.

Rulka, W. (1990) SIMPACK - A computer program for simulation of large motion multibody systems. In W. Schiehlen (Ed.), *Multibody systems handbook* (pp. 265-284). Springer-Verlag: Berlin, Germany.

Seth, A., Sherman, M., Reinbolt, J.A., Delp, S.L. (2011) OpenSim: a musculoskeletal modeling and simulation framework for in silico investigations and exchange. *Procedia IUTAM*, 2(22), pp. 212-232.

Silva, M.T. (2003) *Human motion analysis using multibody dynamics and optimization tools.* PhD Thesis, Technical University of Lisbon, Lisbon, Portugal.

Thelen, D.G. (2003) Adjustment of muscle mechanics model parameters to simulate dynamic contractions in older adults. *Journal of Biomechanical Engineering*, 125(1), pp. 70-77.

Thelen, D.G., Anderson, F.C., Delp, S.L. (2003) Generating dynamic simulations of movement using computed muscle control. *Journal of Biomechanics*, 36(3), pp. 321-328.

Wojtyra, M. (2003) Multibody simulation model of human walking. *Mechanics Based Design of Structures and Machines: An International Journal*, 31(3), pp. 357-379.

# 8

# Concluding remarks



With the intent of developing a computational model able to describe how the biologic structures of the knee joint interact to generate movement and provide stability to the whole body, a multibody approach for contact dynamics was proposed. In this research work, specific concepts and methodologies related to the process of modeling multibody systems were studied. Furthermore, general issues associated with the efficiency of computational methods that deal with 3D-contact events between freeform surfaces were investigated. In short, a comprehensive study on dynamics of multibody systems with contact was offered, giving a special attention to the knee joint modeling and to the process of contact analysis. The main conclusions of this research work were presented throughout this dissertation, being the most important highlighted here. In this Chapter, some suggestions of future developments are also provided.

## 8.1 Conclusions

Within this work, a literature review of the biomechanical models of the knee joint was offered, in which the knee models were classified into two groups: phenomenological- and anatomically-based models. By definition, the anatomically-based models described the mechanical behavior of distinct biologic tissues that surround the knee joint system. As result, these models comprised accurate descriptions of the geometry and material properties of the knee components. The anatomically-based models were divided in three groups: kinematic, quasi-static and dynamic. In the literature survey, the knee joint models were characterized by further

modeling parameters besides the model category, such as the spatial dimensions of the model, i.e. two- or three-dimensional, the mathematical approach, that is, multibody system, finite element method or hybrid, and the modeling joints, which are tibiofemoral joint, patellofemoral joint or both. In order to understand the knee mechanics and comprehend its ability for ensuring two almost mutually exclusive conditions such as stability and mobility, an overview of the human knee joint was provided. This synopsis included the description of the structural anatomy of the knee and of the main movements of this articulation. A brief explanation of the mechanical behavior of the principal biologic tissues that surround the knee joint was also offered. Furthermore, knee joint pathologies and replacement systems were revised.

The multibody formulation of the equations of motion of general biomechanical systems was studied, as well as its numerical solutions. The concept of multibody system was introduced, being the different types of coordinates and kinematic constraints revised. Methodologies to overcome the constraint violations problem were presented, giving a special attention to Baumgarte stabilization method. With the purpose of demonstrating the advantage of using this stabilization method, a multibody model of the human body was developed and utilized as an example of application.

The contact-impact problems and the methodologies utilized to solve and analyze those were investigated. An overview of the existing techniques for geometric detection of contact events was offered. The most common elastic and dissipative laws used to evaluate normal contact forces were studied. Within this dissertation, the modeling process of the contact-impact events was presented in two stages: (*i*) contact detection and (*ii*) contact response. The contact detection is a two-step procedure that includes the identification of the coordinates of the potential contact points and the evaluation of the indentation between the two bodies. In turn, the contact response consists of calculating the contact force based on the state variables of the system and the material properties of the contacting bodies. The dynamic response of the system is obtained by including updated forces into the equations of motion. The application of a method for contact detection purposes depends on the geometrical representation of the contact surfaces. The geometry of a contact surface can be described by using polygonal or non-polygonal models, such as CSG, implicit and parametric approaches. Besides, the contact detection methods are distinguished in broad-, narrow- and single phase approaches. The broad-phase methods identifies smaller groups of objects that may be

colliding and quickly excludes those that definitely are not. Then, narrow-phase methods are employed, which ones test with more accuracy the subgroup of possible contacting pairs pointed out by the broad-phase algorithms. The single-phase formulations are based on the common-normal concept and are used when the simulation only requires a small number of contact calculations in each time interval, as it is the case of the knee joint. The process of evaluating contact forces relies upon appropriate constitutive laws that take into account material and geometric properties of the contacting bodies and, eventually, the impact velocity.

Within this dissertation, a two-dimensional multibody model of the human knee was presented. The development of this knee joint model entails three modeling tasks, namely: (*i*) geometrical representation of contacting profiles by the use of curve fitting techniques based on spline interpolation schemes that keep the geometric convexity; (*ii*) development of a methodology for the accurate prediction of the location of the contact points between two contacting bodies with freeform convex profiles; (*iii*) mathematical description of the nonlinear behavior of the ligaments by a quadratic stress-strain relation. The bones were modeled as perfectly rigid, due to their higher stiffness when compared with the hyaline cartilage, which was considered to be a deformable structure with specific material characteristics. The motion of the tibia relative to the femur was not modeled with conventional kinematic joint, but rather in terms of the action of the ligaments and potential contact between the bones. The proposed contact methodology is based on the common-normal concept and allows for the evaluation of the contact forces generated at the knee contact zone. These contact forces, together with the forces produced by the ligaments, were introduced into the system's equations of motion as generalized forces. Within this work, modeling features associated with contact-impact events were investigated, namely the constitutive law applied to compute contact forces, the convexity of the contact geometries and the presence of a double contact layer. The study of the influence of these variables on the dynamic response of the human knee required two additional modeling steps: (*i*) inclusion of distinct formulations that allow for the detection of contact points whenever two spherical bodies in contact present conformal or non-conformal configurations; (*ii*) implementation of a double layer-based contact model that allows for calculating contact forces on the surface and subsurface.

Several computational simulations using the developed 2D-model of the knee joint were performed, being the dynamic results discussed. These results revealed that

the amplitude of the external applied force has a great influence on the contact forces and, hence, on the ligament forces. It was also shown that the knee medial compartment, which has a conformal configuration, presents higher contact forces when compared with the knee lateral compartment. This observation can explain the major incidence of Osteoarthritis at the medial compartment of the knee. In what concerns with the constitutive contact force models, the approaches proposed by Gonthier *et al.* (2004), Zhiying and Qishao (2006) and Flores *et al.* (2011) demonstrated to be reasonable options to compute the knee contact forces, since they describe the nonlinear behavior of the hyaline cartilage and also take into account its damping properties typical of inelastic materials. Regarding the materials of the knee contact interface, the results showed that the hyaline cartilage is an outstanding shock absorber and load spreader since it reduces the contact force and extends the period of contact. It was also concluded that for higher contact indentations, an artificial knee produces lower contact forces than a pathologic knee that is destitute of almost all of the articular cartilage. This suitable dynamic response of the artificial knee is due to the elastic and damping properties of the ultra-high molecular weight polyethylene, which is the most used bearing material in joint replacement systems.

A multibody formulation to deal with spatial contact problems was also presented. Within the 3D-approach, the contact surfaces are described by means of point-clouds extracted from parametric representations. This type of geometrical models offers a great flexibility and precision for handling free-form shapes. Moreover, parametric surfaces exhibit continuity between adjacent patches that guarantee numerical stability and allows for the reduction of a 3D-problem to the 2D-domain. With the intent to reduce the computational time, a preprocessor unit was added to the proposed algorithm for spatial dynamic analysis. During the preprocessing technique, the contact surfaces are arranged for the dynamic simulations. The surface preparation consists in organizing the geometric data relevant for contact into a lookup table, which is stored in memory as a direct access file. The preprocessor unit allowed for a significantly reduction of the amount of memory required for data storage and an improvement of the computational efficiency of the contact detection process. The proposed formulation is particularly useful in continuous contact scenarios, such as in the knee joint, where the location of the contact points undergoes slight smooth variations. Therefore, the spatial multibody algorithm was utilized to evaluate the dynamic behavior of the human knee. Similarly to the 2D-case, the 3D-model of the knee joint was composed by two contacting bodies,

the femur and the tibia, which were connected by nonlinear springs that represented the main ligaments of the knee joint. Regarding the contact problem, two contact pairs were considered for modeling the medial and lateral contacts at the knee joint. The femoral condyles were represented using spheres, while the tibial plateaus were modeled as flat surfaces. The first impact was detected at the medial condyle and the second at the lateral side. Afterwards, the tibia remained in contact with femur during the whole simulation, or at lateral compartment, or at medial side or at both. Together with the knee flexion, also studied in the 2D-case, an internal rotation of the tibia was observed. This movement is the typical response of the knee when it flexes under conditions of non-weight bearing. Ligament forces were also analyzed during the 3D-simulations.

A four-step framework of how to build and analyze a knee joint model using OpenSim was offered in this work, being the limitations of using this software pointed out. Several modeling and analysis features available in OpenSim were investigated. Elements such as rigid bodies, joints, markers, muscles and contact surfaces, were explored and included into the model. Four analysis tools were studied, namely inverse kinematics, inverse dynamics, static optimization and computed muscle control. Within this work, it was concluded that OpenSim is suitable for modeling, visualizing and analyzing human movements. Regarding contact analysis, the results showed that OpenSim contact formulation requires further investigation, improvement and validation, mostly when volumetric meshes are used to represent the contact surfaces.

## 8.2 Suggestions of future developments

The methodology proposed throughout this dissertation is general and can be used to analyze the dynamic behavior of a wide range of multibody models: from mechanical linkages to biological systems. Furthermore, the present research work comprised the development/implementation of mathematical models, numerical methods and computational algorithms. As a result, numerous aspects and issues can be pointed out as object of future investigations. In this section, five forthcoming studies are suggested, which are focused on the developed knee joint model utilized as application example.

Knee joint motion is a result of the bony geometry, the soft-tissue structures, the joint loading and the muscle activation. The absence of muscles into the developed knee model is the major shortcoming, since muscles are responsible for the movements and the stability of this human articulation. Hence, musculoskeletal elements have to be

included into the model, as well as a method for the solution of redundant muscle forces in human locomotion based on optimization tools. In future simulations, the patella and the menisci should also be added to the developed knee model.

Other upcoming study is combining the developed knee model with a whole body model, and analyze the dynamic response of this biomechanical model during various activities of daily life, such as walking, running, climbing stairs, etc.

The third suggestion of future work consists of predicting muscle and contact contributions to knee dynamic loads during gait. For this purpose, it is suggested seeking for a mathematical relation able to describe the balance between muscles and contact forces at the knee joint. This investigation aims to obtain accurate information on the contact geometry and muscle properties that can be useful on the design and manufacturing of knee replacement systems.

The low friction and wear of articular layers has been attributed to mixed modes of lubrication which include fluid film lubrication by synovial fluid, lubrication by pressurization of the interstitial fluid of cartilage (weeping) and, boundary lubrication by a variety of candidate molecules in synovial fluid and cartilage. Therefore, the implementation of mixed lubrication modes into the developed multibody algorithm is recommended as future development. The goal of this forthcoming work is to investigate the influence of a lubrication contact law on the dynamics of the knee joint.

Computational predictions of cartilage wear at the knee joint are pointed out as future application. The proposal is to implement the Archard's law into the developed algorithm for contact dynamics. This upcoming study aims to support the diagnosis and clinical treatment of degenerative knee diseases, such as Osteoarthritis.

It is worth noting that experimental data obtained from cadaveric specimens, electromyography, fluoroscopic data, dynamic knee simulator, magnetic resonance imaging, materials and mechanical testing and/or motion trials are very important and should be utilized as much as possible to complement the future studies listed above.