Modeling and Simulation Workflow Using Natural Knee Data

Model Reuse Specifications

Cleveland Clinic Approach

Document created on February 22, 2022; last updated on April 5, 2022.

Document prepared by

Snehal K. Chokhandre, MSc – chokhas@ccf.org, +1 (216) 445 3555

Department of Biomedical Engineering and Computational Biomodeling (CoBi) Core, Lerner Research Institute, Cleveland Clinic, 9500 Euclid Avenue (ND20), Cleveland, OH 44195, USA

Ellen Klonowski, BSc – klonowe@ccf.org , +1 (216) 445 3555

Department of Biomedical Engineering and Computational Biomodeling (CoBi) Core, Lerner Research Institute, Cleveland Clinic, 9500 Euclid Avenue (ND20), Cleveland, OH 44195, USA

***Ahmet Erdemir, PhD –** erdemira@ccf.org, +1 (216) 445 9523

Department of Biomedical Engineering and Computational Biomodeling (CoBi) Core, Lerner Research Institute, Cleveland Clinic, 9500 Euclid Avenue (ND20), Cleveland, OH 44195, USA

**For correspondence.*

Table of Contents

Cite this document as

> Chokhandre, SK, Klonowski E, Erdemir, A, *Modeling and simulation workflow using Natural Knee data: model reuse specifications – Cleveland Clinic approach*, April 5, 2022, Cleveland Clinic, Cleveland, Ohio, USA.

Contributions to this document by Snehal Chokhandre, Ellen Klonowski, and Ahmet Erdemir were supported by the National Institute of Biomedical Imaging and Bioengineering, National Institutes of Health (R01EB024573; *Reproducibility in simulation-based prediction of natural knee mechanics*; Principal Investigator: Erdemir) and in part by the National Institute of General Medical Sciences, National Institutes of Health (R01GM104139; *Open Knee(s): virtual biomechanical representations of the knee joint*, Principal Investigator: Erdemir).

Synopsis

This document describes planned model reuse specifications that are aimed at utilizing knee joint model generated for a specimen from Natural Knee data set[1] in *Model Benchmarking* phase[2], to demonstrate utility of models in cases for which the models are not specifically developed and do not have experimental data available for validation. The proposed modeling activities are in response to the *Model Reuse* phase[3] of the project *Reproducibility in simulation-based prediction of natural knee mechanics*, a study funded by the National Institute of Biomedical Imaging and Bioengineering, National Institutes of Health (Grant No. R01EB024573)[4]. The outlined choices for modeling and simulation processes represent those of the Cleveland Clinic team. These choices are primarily aimed for pragmatic, yet comprehensive, reuse of an anatomically and mechanically detailed and extensible knee joint model incorporating its major tissue structures.

Benchmarked Model

Described model reuse workflow utilizes one of the models generated in the *Model Benchmarking* phase[2], of the project *Reproducibility in simulation-based prediction of natural knee mechanics*, a study funded by the National Institute of Biomedical Imaging and Bioengineering, National Institutes of Health (Grant No. R01EB024573)[4]. This model was based on an existing data set from the Natural Knee Data, specifically those from specimen DU02[1]. Development of this model was described as part of the model development specifications[5], model calibration specifications[6] and model benchmarking specifications[7]. The *Model Reuse* workflow will utilize the benchmarked model with optimal mesh densities obtained via mesh convergence studies and, material properties in the *Model Calibration* phase[8] and recalibrated in situ strain values from *Model Benchmarking* phase[9]. The benchmarked model i registered to experimental coordinate system.

Overview of Modeling and Simulation Processes

A previously developed and benchmarked three-dimensional computational model of the knee, specifically finite element representation of the tibiofemoral and patellofemoral joints, will be utilized for reuse demonstration. Knee model registered to experimental coordinate system, with optimal mesh densities, material properties and in situ strain values, will be used. Specimen-specific experimental kinetics-kinematics data is not available for the Reuse case simulation conditions. Specific goals for this phase will be to conduct simulations for passive flexion loading, pivot shift loading and weight bearing X-ray loading.

FEBio[10], along with FEBio PreStrain Plugin[11], will be used to conduct finite element analysis (solid mechanics, based on implicit static solver). Simulation results will be visualized using PostView[12]. Specimen-specific joint mechanics data and predicted kinematics-kinetics of the joints will be processed to report joint movements in all loading cases. Python[13] and SciPy[14], along with auxiliary Python packages, will be used to automate data analysis, model customization, and post processing. All modeling and simulation outputs, intermediate and final, will be publicly disseminated through an online repository[15].

Detailed Modeling and Simulation Outputs

Model Reuse specifications will result in the following intermediate and final outputs. The Workflow section below provides detailed instructions on how to obtain these.

File	Description	File Format
Model Properties	XML based text file which specifies the material properties of all tissues in the model, and the coordinates of the manually chosen anatomical landmarks (used as input for the customization script)	.xml[16]
Customized Models (FEBio Input File)	XML based text file (for finite element analysis with FEBio) customized to include mesh definitions, tissue interactions, tissue-specific constitutive models, in situ ligament strains, representation of additional stabilizing structures, anatomical knee joint coordinate systems, specialized loading and boundary conditions and numerical analysis settings to represent customizations for reuse case loading scenarios.	.feb[17]
Raw Simulation Results	Binary (.xplt) and text files (.log) obtained by simulation of reuse cases	.xplt[12] .log[17]
Processed Simulation Results	CSV based text files storing extracted knee kinematics and kinetics during reuse simulations; processed using raw simulation results and supported by graphs as binary image files	.csv[18] .png[19]

Workflow

Customized Full Models for Reuse Simulations

Target Outcome

Customized full knee models in FEBio[10] format (.feb[17], XML[16] based text file) prepared for all simulation cases; models will include converged meshes, confirmed material properties, calibrated in situ ligament strains and experimental coordinate system.

Burden

Software requirements:

Python. Python is a high-level multi-platform programming language (free and open source GPL compatible Python Software Foundation license, see https://www.python.org)[13]. Any contemporary version available for the computing platform can be used; 3.8.0 is the latest version at the time of preparation of this document. Depending on the requirements of legacy Python scripts, version 2.7 may be used.

SciPy. SciPy is a Python based open source software platform for mathematics, science and engineering (free and open source BSD-new license, see https://www.scipy.org)[14]. Any contemporary version available for the computing platform can be used; 1.3.1 is the latest version at the time of preparation of this document. Depending on the requirements of legacy Python scripts, a version compatible with Python 2.7 may be used.

Python Scripts. There are existing Python scripts developed in *Model Calibration* phase[20]. Latest editions can be found at the source code repository at https://simtk.org/svn/openknee/app/KneeHub/src/. Reuse loading cases will be generated using in house Python script experiment_to_model.py developed in the *Model Calibration* phase. This script allows updating the registered models (obtained using FebCustomization_p3.py) with the appropriate experimental loading. For reuse cases, desired loading will be represented in a manner similar to the experimental loading, as described later in the document. The registered models are customized with converged meshes and confirmed material properties and calibrated in situ ligament strains and experimental joint coordinate system. For ACL deficient model, previously developed Python script ModelReduction_rigids.py will be used. The script converts the component to be removed (in this case ACL) to a rigid body and removes all ties and contacts.

<u>Hardware requirements:</u>

Any contemporary computer; desktop, workstation, or laptop. All aforementioned software are supported on multiple platforms including Windows, Mac OS X, and Linux.

<u>Anticipated Man Hours and Expertise Level:</u>

1 day of full-time effort from a research engineer with Master's degree, mechanical/biomedical background, <3 years of research experience, and familiarity to finite element analysis and scripting.

<u>Computational Cost:</u>

Minimal compared to required interactions with computer scripts.

Protocols

Input

Template FEBio model file of the full knee (.feb)[17] and model properties (.xml)[16] files for with converged meshes, confirmed material properties, experiment coordinate systems, calibrated in situ ligament strains and target <u>kinetics- kinematics of simulated trajectories</u>.

Customization for Reuse Simulation Cases

Customization scripts developed for *Model Calibration* phase[8], in particular experiment_to_model.py will be used to generate models representative of the reuse simulation cases. Loading and boundary conditions and output requests will be the same, as described in Model Development specifications and are briefly summarized in here. <u>Tibia will be fixed; femur and patella will be free to move.</u> From time 0 to 1, in situ strain will be applied while keeping flexion at 0°. From time 1 to 2, the loads and boundary conditions at the start of reuse case will be prescribed, i.e., the flexion angle will be set and the loads in the remaining degrees of freedom will be applied on femur. From time 2 to 3, the loads and boundary conditions of the reuse trial will be applied until the end point. Load curves for each degrees of freedom (particularly the dominant loading) will be defined based on expected simulation scenarios (passive flexion, pivot shift, weight bearing X-ray) and simulation output will be requested at each data point. The kinematics kinetics trajectories of reuse cases will be split to facilitate prescription of loading scenarios in simulations. Kinetics will be applied to the femur in image coordinate system, kinematics are in joint coordinate system. A total of 4 models will be generated:

1. 90° passive flexion prescribed at femur fixed flexion axis with body weight compressive joint force applied on tibia fixed superior – inferior axis. No additional forces or moments will be applied. Remaining rotation and translations will be free. *Target predictions:* knee joint kinematics- kinetics.
2. Pivot shift simulation with knee loaded by prescribed flexion from 0° to 90° while simultaneously applying and internal rotation torque of 5 Nm and valgus torque of 10 Nm. Flexion will be prescribed at femur fixed flexion axis. Internal rotation torque will be applied on femur at a tibia fixed superior-inferior axis. Valgus torque will be applied on femur at a tibia fixed anterior-posterior axis. *Target predictions:* knee joint kinematics-kinetics from 0°to 90°, clinical anterior translation of tibia along tibia fixed anterior axis and, prediction of ACL forces and strains.
3. Pivot shift simulation with ACL deficient knee loaded by prescribed flexion from 0° to 90° while simultaneously applying and internal rotation torque of 5 Nm and valgus torque of 10 Nm. Flexion will be prescribed on femur at a femur fixed flexion axis. Internal rotation torque will be applied on femur at tibia fixed superior-inferior axis. Valgus torque will be applied on femur at a tibia fixed anterior-posterior axis. *Target predictions:* knee joint kinematics-kinetics from 0°to 90° and, clinical anterior translation of tibia along tibia fixed anterior axis.
4. Wight-bearing x-ray simulation with knee loaded by prescribed flexion from 0° to 15° degree and ½ body weight. Flexion will be prescribed on femur at a femur fixed flexion axis. ½ body weight compressive joint force will be applied on femur at tibia fixed superior-inferior axis. *Target predictions:* Cartilage contact stresses. Optionally, medial lateral contact forces.

Simulations

Target Outcome

Solutions of customized full knee models through finite element analysis using FEBio[17]; generating simulation results as binary and text output files (.xplt and .log, respectively)[10].

Burden

Software requirements:

FEBio. FEBio[10] is a nonlinear implicit finite element analysis framework designed specifically for analysis in biomechanics and biophysics (binaries custom open source license; free for academic research use, licensing for commercial use is available, see http://www.febio.org)[10]. Version 2.9 will be used.

FEBio PreStrain Plugin. PreStrain[11] Plugin provides a general framework for representing prestrain in a finite element model using a prestrain gradient method. The version used for the *Model Benchmarking phase*[2] will be used.

Hardware requirements:

Any contemporary computer; desktop, workstation, or laptop. All aforementioned software are supported on multiple platforms including Windows, Mac OS X, and Linux. Access to a high performance computing cluster can expedite simulations by running multiple finite element analysis cases in parallel.

Anticipated Man Hours and Expertise Level:

3 days of full-time effort from a research engineer with Master's degree, mechanical/biomedical background, over 3 years of research experience, and familiarity to finite element analysis.

Computational Cost:

~6 hours of anticipated simulation time (wall clock) per simulation case; a total of 4 simulation cases.

Protocols

Input

Customized full models in FEBio format (.feb[17]).

Simulation Process

Invoke FEBio[17] with each customized model file as input. If a simulation does not convergence, convergence tolerances and utilization of alternative solution algorithms may need to be employed in a fashion similar to iterations conducted during the *Model Development phase*[21], *Model Calibration* phase[8] and *Model Benchmarking* phase[2].

Post-Processing

Target Outcome

Extraction and summary of knee kinematics and kinetics of all simulation cases as text based files (.csv[18]); processed using raw simulation results of customized models with FEBio (.log file[17]), supported by graphs as binary image files (.png[19]). Additionally, extraction and summary of cartilage contact stresses in weight bearing X-ray simulation, and anterior translation of tibia along tibia fixed anterior axis during pivot shift simulations.

Burden

Software requirements:

Python. Python[13] is a high-level multi-platform programming language (free and open source GPL compatible Python Software Foundation license, see https://www.python.org)[13]. Any contemporary version available for the computing platform can be used; 3.8.0 is the latest version at the time of preparation of this document. Depending on the requirements of legacy Python scripts, version 2.7 may be used.

SciPy. SciPy is a Python based open source software platform for mathematics, science and engineering (free and open source BSD-new license, see https://www.scipy.org)[14]. Any contemporary version available for the computing platform can be used; 1.3.1 is the latest version at the time of preparation of this document. Depending on the requirements of legacy Python scripts, a version compatible with Python 2.7 may be used.

Python Scripts. Post processing will be performed using script LogPostProcessing.py, described previously, for extracting joint kinematics and kinetics from the model outputs. The script will be updated to process and extract cartilage contact stresses and obtaining anterior translation of tibia along tibia fixed anterior axis. Source code available at https://simtk.org/svn/openknee/app/KneeHub/src/

PostView. PostView[12] is a post-processor to visualize and analyze results from FEBio[17], finite element analysis package for biomechanics (binaries custom open source license; free for academic research use, licensing for commercial use is available, see https://febio.org/postview/). The version used for the *Model Benchmarking* phase[2] will be used.

Hardware requirements:

Any contemporary computer; desktop, workstation, or laptop. All aforementioned software are supported on multiple platforms including Windows, Mac OS X, and Linux.

Anticipated Man Hours and Expertise Level:

2 days of full-time effort from a research engineer with Master's degree, mechanical/biomedical background, >3 years of research experience, and familiarity to finite element analysis and scripting.

Computational Cost:

Minimal compared to required interactions with computer scripts.

Protocols

Input

Solutions (simulation results) of customized full models through finite element analysis using FEBio[17] as binary and text output files (.xplt and .log, respectively)[10]; processed experimental knee kinematics and kinetics as text files (.csv[18]).

Standalone Processing of Simulation Results

A Python script previously developed in the *Model Development* phase[20](LogPostProcessing.py) will be used to read the log file and extract, store (as .csv[18]), and plot knee kinematics and kinetics during all simulation cases (as .png[19]) for both tibiofemoral and patellofemoral joints. Cartilage contact stresses will be extracted manually from Postview[12].

Visualization

PostView[12] will be used to take snapshots of the model at different flexion angles, as obtained through simulation of passive flexion. PostView[12] can also be used to inspect tissue stress-strain distributions, export data, images, and animations.

Dissemination

Target Outcome

Modeling and simulation outputs delivered to the public as a download package.

Burden

Infrastructure:

SimTK. SimTK is a free project-hosting platform for the biomedical computation community (see https://simtk.org/)[15]. Project sites at SimTK provide source code repositories, wikis to support development; and news, forums, downloads and documents sections to engage with user communities.

Anticipated Man Hours and Expertise Level:

Less than a day of full-time effort (to prepare, organize, and disseminate final package) from a research engineer with Master's degree, mechanical/biomedical background, > years of research experience, and familiarity to public dissemination.

Protocols

All modeling and simulation outputs of the *Model Reuse* phase[3] will be collated as a package and uploaded to the project site of *Reproducibility in simulation-based prediction of natural knee mechanics* located at SimTK[15] (https://simtk.org/projects/kneehub/). This download package will be accessible by the public licensed under Creative Commons Attribution 4.0 International License.

Protocol Deviations

Target Outcome

Protocol deviations to model benchmarking specifications documented and delivered to the public.

Burden

Infrastructure:

SimTK. SimTK[15] is a free project-hosting platform for the biomedical computation community (see https://simtk.org/). Project sites at SimTK provide source code repositories, wikis to support development; and news, forums, downloads and documents sections to engage with user communities.

Anticipated Man Hours and Expertise Level:

For each protocol deviation, on the order of minutes of full-time effort, and for final report, less than a day of full-time effort, from a research engineer with Master's degree, mechanical/biomedical background, >3 years of research experience, and familiarity to finite element analysis.

Protocols

It is anticipated that some deviations to modeling and simulation workflow, described in here as part of *Model Reuse* phase[3], will happen. There is also the possibility that some information on model reuse may be missing. All these will be documented on an ongoing basis during the execution of the planned workflow. Final document will be submitted to the project site of *Reproducibility in simulation-based prediction of natural knee mechanics* located at SimTK (https://simtk.org/projects/kneehub/) as a publicly accessible document under Creative Commons Attribution 4.0 International License.

Overall Burden

Overall burden of the modeling and simulation workflow described in here is determined by the requirements for data, labor, software and hardware, and other infrastructure. Software and hardware costs are associated with pre-/post-processing of simulations in a coherent manner. It is anticipated that the model reuse and analysis of simulation results in light of experimental joint mechanics data can be performed in any contemporary computer, minimizing hardware costs. All software packages used in the modeling and simulation workflow are freely available: Python[13] and SciPy[14] – to utilize Python scripts (existing and some to be updated) for pre- and post-processing of models; FEBio[17] and FEBio PreStrain Plugin[11] – for finite element analysis; and PostView[12] – for visualization of simulation results. The activity will leverage SimTK[15] for public dissemination. SimTK[15] is a freely available project hosting site for biomedical computing. Labor effort will be at a minimum of 2 weeks of full time effort from a research engineer with Master's degree, mechanical/biomedical background, more than 3 years of research experience, and familiarity to finite element analysis. This effort level includes all modeling activities, record keeping, and dissemination. It should be noted that this estimate relies on the assumption that modeling and simulation processes complete as planned, without any significant deviations and iterations. Based on our recent experience in the *Model Benchmarking* phase[9], convergence problems may require iterative troubleshooting of simulations. High simulation cost (~6 hours for passive flexion) may also be a confounding factor. As a result, this timeline may extend in an agile fashion. The overall burden of the model reuse specifications should be evaluated in concert with their desired final outcome – a comprehensive and extensible knee joint model incorporating anatomical and mechanical detail of its major structures, which is capable of demonstrating utility of such models without the supporting specimen-specific experimental mechanical response.

References

1. Natural Knee Data | Center for Orthopaedic Biomechanics | University of Denver. https://digitalcommons.du.edu/natural_knee_data/.
2. ModelBenchmarking - kneehub. https://simtk.org/plugins/moinmoin/kneehub/ModelBenchmarking.
3. ModelReuse - kneehub. https://simtk.org/plugins/moinmoin/kneehub/ModelReuse.
4. SimTK: Reproducibility in simulation-based prediction of natural knee mechanics: Project Home. https://simtk.org/projects/kneehub.
5. Schwartz, A., Chokhandre, S. & Erdemir, A. *Modeling and simulation workflow using Natural Knee Data: model development specifications – Cleveland Clinic approach*. (2018).
6. Schwartz, A., Chokhandre, S. & Erdemir, A. *Modeling and simulation workflow using Natural Knee data: model calibration specifications - Cleveland Clinic approach*. (2019).
7. CC-NKD-MB-specifications.pdf.
8. ModelCalibration - kneehub. https://simtk.org/plugins/moinmoin/kneehub/ModelCalibration.
9. ModelBenchmarking - kneehub. https://simtk.org/plugins/moinmoin/kneehub/ModelBenchmarking.
10. FEBio Software Suite. https://febio.org/.
11. Maas, S. A., Erdemir, A., Halloran, J. P. & Weiss, J. A. A general framework for application of prestrain to computational models of biological materials. *Journal of the Mechanical Behavior of Biomedical Materials* **61**, 499–510 (2016).
12. PostView – FEBio Software Suite. https://febio.org/postview/.
13. Welcome to Python.org. *Python.org* https://www.python.org/.
14. SciPy.org — SciPy.org. https://www.scipy.org/.
15. SimTK: Welcome. https://simtk.org/.
16. Office Open XML - What is OOXML? http://officeopenxml.com/.
17. FEBio User Manual Version 2.9. https://help.febio.org/FEBio/FEBio_um_2_9/index.html.
18. Shafranovich, Y. Common Format and MIME Type for Comma-Separated Values (CSV) Files. (2005).
19. Roelofs, G. PNG: The Definitive Guide. (1999).
20. ModelDevelopment - kneehub. https://simtk.org/plugins/moinmoin/kneehub/ModelDevelopment.