**Kinematic Calibration Documentation**

Kinematic calibration is a free software package for kinematic model calibration. The goal is to run optimizations in Matlab that adjust joint parameter values in a subject-specific OpenSim model until an inverse kinematics analysis achieves the best possible match to experimentally measured surface marker locations.

**Input Setup File Description**

The input XML file allows the user to setup the entire kinematic calibration routine. The file consists of three main components, or high-level tags, and an array of sub-tags that allow the user to define various aspects of the calibration routine.

**High-level Tags**

Joint parameters tag < JointParameters>: Allows the user to define the joint design parameters that are to be included in the optimization

Marker plate parameters tag <MarkerPlateParameters>: Allows the user to define the marker plate joint design parameters that are to be included in the optimization

Input settings tag < InputSettings>: Allows the user to define the input OpenSim model filename, output (optimized) OpenSim model filename, inverse kinematics solver accuracy, scale set setup file for rescaling purposes, and optimization groupings that define a range of settings that include name of file containing marker locations, simulation time range, optimization order, etc.

For complete details please refer to the example below and the provided input setup file (inputSetupFile.xml) located in the **examples** folder.

**Joint and Marker Plate Parameter Setup Examples**

As remarked above, the joint and marker plate parameter tags set the optimization design parameters for joints and marker plates, respectively. Specifically, these tags define which of the parameters in the location/orientation in child/parent should be included in the optimization. In addition, the maximum allowable movement/angle change for each of the selected parameters can be defined within these tags.

**Example 1** Right hip joint parameter setup

In this example, the y and z, i.e., the vertical and medial-lateral, components of the location in parent of the right hip joint are to be included in the optimization routine with maximum allowable movement change of 1 cm in both directions.

**<JointParameters name="rightHipJoint">**

**<apply>true</apply>**

**<joint\_name>hip\_r</joint\_name>**

**<joint\_type>Custom</joint\_type>**

**<location\_in\_parent\_params>0 1 1</location\_in\_parent\_params>**

**<location\_in\_parent\_maxParamChange>0 0.01 0.01</location\_in\_parent\_maxParamChange>**

**</JointParameters>**

**Example 2** Right tibia marker plate parameter setup

In this example, the x, y, and z, i.e., the anterior-posterior, vertical, and medial-lateral, components of both the location and orientation in parent of the right tibia marker plate joint are to be optimized with maximum allowable movement and angle change, in all directions, of 10 cm and 1 deg., respectively.

**<MarkerPlateParameters name="rightTibiaMarkerPlate">**

**<apply>true</apply>**

**<marker\_plate\_name>tibia\_r\_markerPlate\_jnt</marker\_plate\_name>**

**<marker\_plate\_joint\_type>weld</marker\_plate\_joint\_type>**

**<location\_in\_parent\_params>1 1 1</location\_in\_parent\_params>**

**<location\_in\_parent\_maxParamChange>0.1 0.1 0.1</location\_in\_parent\_maxParamChange>**

**<orientation\_in\_parent\_params>1 1 1</orientation\_in\_parent\_params>**

**<orientation\_in\_parent\_maxParamChange>5 5 5</orientation\_in\_parent\_maxParamChange>**

**</MarkerPlateParameters>**

**Input Settings Example**

The input settings contains sub-tags that defined the *name of the input OpenSim model*, the *name of the output (optimized) model*, the *inverse kinematics solver accuracy*, and the *name of the scale set setup XML file* that contains a set of scale factors all set to 1. This file is used in the rescaling portion of the kinematic calibration routine, which ensured that the correct scaling factors are reflected in the optimized model.

In addition to these, other sub-tags are defined within the input settings tag that allows the user to specify how the various optimizations in the kinematic calibration routine are to be executed. Specifically, for each optimization group (<OptimizationGroup>) the following information can be specified: *name of file containing maker locations*, *name of coordinate file containing pre-computed joint angles*, the *simulation time range*, *inverse kinematics error flag*, *name of inverse kinematics output motion file*, i.e., the file containing predicted joint angles, *inverse kinematics marker locations report flag*, *name of the inverse kinematic tasks setup file* containing marker and coordinate (if applicable) weight factors, the *order* in which the current optimization should be executed, the *joints* and *maker plates* that are to be included in the optimization, *flag indicating if the optimization should penalize changes in the design parameters*, and if the optimization should enforce joint symmetry (translation/orientation), and if so, in which direction (tx/rx, ty/ry, and/or tz/rz). All of these settings are defined within the <OptimizationGroupings> tag.

**Example 3** Knee optimization input settings setup

In this example, a pair of optimizations is executed that includes the right/left knee joints and right/let shank marker plates, and after the optimization is over, the right/left femur is rescaled along the vertical, i.e., y, direction.

**<InputSettings>**

**<input\_model\_file>inModel.osim</input\_model\_file>**

**<accuracy>1e-7</accuracy>**

**<output\_model\_file>optModel.osim</output\_model\_file>**

**<scale\_set\_file>ScaleSetSetupFile.xml</scale\_set\_file>**

**<OptimizationGroupings>**

**<OptimizationGroup name="knee">**

**<marker\_file>expData.trc</marker\_file>**

**<time\_range>0.0 2.45</time\_range>**

**<report\_errors>true</report\_errors>**

**<output\_motion\_file>jointAngles.mot</output\_motion\_file>**

**<report\_marker\_locations>true</report\_marker\_locations>**

**<ik\_set\_file>IKTasksSetupFile.xml</ik\_set\_file>**

**<optimization\_order>1</optimization\_order>**

**<WhichJoints>**

**<joint\_name>knee\_r</joint\_name>**

**<joint\_name>knee\_l</joint\_name>**

**</WhichJoints>**

**<WhichMarkerPlates>**

**<marker\_plate\_name>tibia\_r\_markerPlate\_jnt</marker\_plate\_name>**

**<marker\_plate\_name>tibia\_l\_markerPlate\_jnt</marker\_plate\_name>**

**</WhichMarkerPlates>**

**<penalize\_changes>false</penalize\_changes>**

**<scale\_segments>true</scale\_segments>**

**<scale\_directions>0 1 0</scale\_directions>**

**</OptimizationGroup >**

**</OptimizationGroupings>**

**</InputSettings>**

**List of Source Codes (…\src)**

**addMarkerPlatesToModel.m**: Function adding marker plates to an OpenSim model

**combineTrcFile.m**: Function combining two or more .trc files into one single .trc file

**costFunction.m**: Optimization cost function

**kinCal.m**: Main kinematic calibration program

**utilFunctions.m**: Utility functions used in the main kinematic calibration program

**xmlRead.m**: XML file reader (3rd party code)

**Calibration Flow**

1. Calibrate Torso, Pelvis, and Femur marker plate locations in parent on both sides simultaneously
2. Calibrate Knee joint parameters and Tibia plate location in parent on both sides independently
3. Scale Thigh segment in Superior-Inferior direction
   1. Scaling in other directions is not advisable
   2. Be certain to reset marker plate locations after scaling!
4. Calibrate Ankle and Subtalar axes parameters on both sides
5. Scale Tibia length
   1. Based on new distance from Knee joint to Ankle joint
   2. Scaling in other directions is not advisable
   3. Be certain to reset marker plate locations after scaling!
6. Set Ankle and Subtalar offsets such that the zero position is more realistic
   1. After parameter scaling, physiological joint angles may be close to bounds in OpenSim model, leading to IK errors when applying model to new data
   2. Reorienting the Talus for cosmetic purposes
7. Calibrate all of the parameters that can be optimized simultaneously, while penalizing parameters for changing too far from their initial values
   1. Scaled dv, i.e., delta change in design parameters, are added to the cost function to penalize changes in model, i.e., design, parameters. Errors from these terms will be 1 when **dv = maxParamChange** listed in input setup file
   2. Scaling after this step may not be necessary, since parameters change only minimally during this phase of the calibration

**Recommendations**

To be included…