

# **An OpenSim plugin to extract the muscle lines of action and attachments.**

**Author: Luca Modenese**

**[l.modenese08@imperial.ac.uk](mailto:l.modenese08@imperial.ac.uk)**

[www.imperial.ac.uk/structuralbiomechanics](http://www.imperial.ac.uk/structuralbiomechanics)

---

**STRUCTURAL BIOMECHANICS**  
in the Department of Civil and Environmental Engineering

Use of the 'MuscleForceDirection' plugin in OpenSim is only permitted if the following conditions are met:

1. The software is used only for non-commercial research and education. It may not be used in relation to any commercial activity.
2. Use of the plugin must be acknowledged in all publications, presentations, or documents describing work in which this plugin is used to obtain results.
3. Credits to developers may not be removed from executables created from modifications of the source.
4. The user agrees to the usage terms as provided by OpenSim: THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR BUSINESS INTERRUPTION) OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**Please cite the following reference:**

Modenese L., 2012. *Hip contact force prediction using a musculoskeletal model of the lower limb*. PhD Thesis, Imperial College London, London.

**SimTK related project:** [https://simtk.org/home/force\\_direction](https://simtk.org/home/force_direction)

## Contents

1 Introduction .....	1
1.2 Anatomical and effective muscle attachments .....	2
2 The MuscleForceDirection plugin .....	3
2.1 How to install the plugin .....	3
2.2 Inputs and Outputs .....	3
2.3 Using the MuscleForceDirection analysis from the GUI .....	4
2.3.1 The properties in the MuscleForceDirection analysis panel .....	5
2.3.2 Setup file .....	6
2.4 Using the MuscleForceDirection analysis from the command line .....	6
2.5 Calling the MuscleForceDirection analysis from Matlab .....	7
2.6 The output files .....	7
2.7 How MuscleForceDirection works .....	8
3 Acknowledgments.....	8

# 1 Introduction

This plugin was implemented with the aim of extending the functionality of OpenSim in order to extract from musculoskeletal models useful information for setting up finite element analyses. In particular, given an appropriate musculoskeletal model e.g. the one in Figure 1 (A), it is desirable to be able to extract:

1. the muscle attachment positions
2. the muscle force directions

in order to apply external loads representative of the muscle forces to the bone mesh, as in the finite element analysis shown in Figure 1 (B). In the current OpenSim version, a straightforward way of extracting the muscle attachments and force direction is not available and the purpose of this plugin is to implement a new analysis to retrieve that information.

The plugin included in this package implements a new analysis called `MuscleForceDirection` that allows the user to extract the attachment positions and lines of action of the muscles attached to the selected bodies for an assigned kinematics. Both the attachments and the muscle force directions can be expressed in the ground or in the segment reference frame.

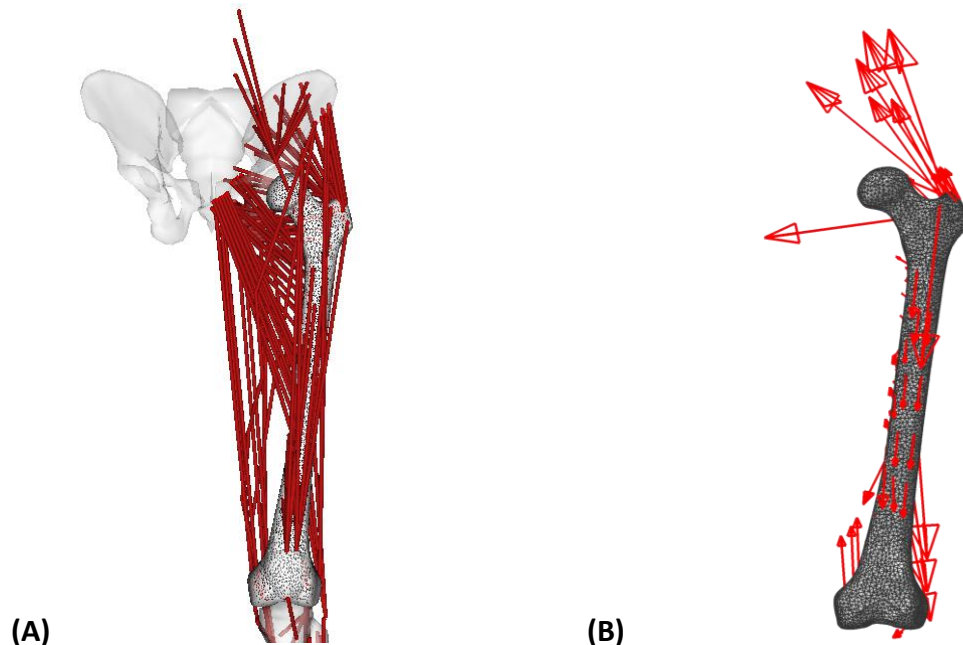


FIGURE 1 THE AIM OF THIS PLUGIN IS TO HELP OPENSIM USERS IN EXTRACTING INFORMATION FROM MUSCULOSKELETAL MODELS (A) FOR USE IN FINITE ELEMENT MODELS (B)

## 1.2 Anatomical and effective muscle attachments

When dealing with muscle attachments and muscle lines of action is important to appreciate the distinction between anatomical and effective muscle attachments.

The anatomical attachments are the locations where the muscles are directly connected to the bone surface, but the direction of the muscle force associated with this point is not necessarily representative of the mechanical effect of that muscle on the segment if via points or wrapping surfaces influence the muscle path, as Figure 2 demonstrates for *gastrocnemius medialis*.

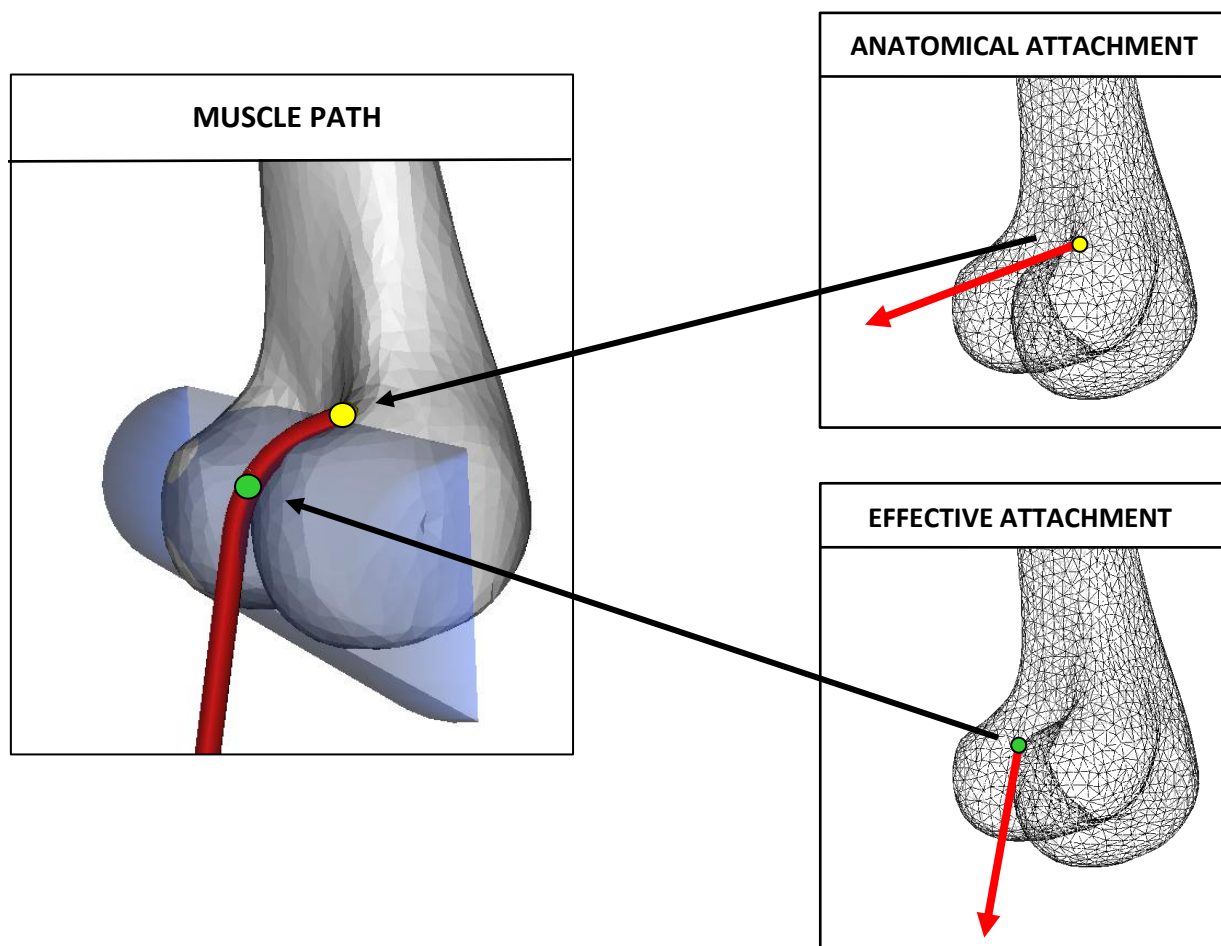


FIGURE 2 FOR A GIVEN MUSCLE PATH THE ANATOMICAL (YELLOW DOT) AND EFFECTIVE (GREEN DOT) FEMORAL ATTACHMENTS ARE REPRESENTED, TOGETHER WITH THE ASSOCIATED MUSCLE DIRECTION (RED ARROW).

## 2 The MuscleForceDirection plugin

### 2.1 How to install the plugin

The following steps will guide the user through the installation of the plugin:

1. If you do not have any other plugin already installed, create a new folder called “plugins” in your OpenSim installation directory. Typically, for OpenSim 2.4.0 the folder will be “C:\OpenSim2.4.0\plugins”.
2. Copy the file “MuscleForceDirection.dll” to the “plugins” folder.
3. Load the plugin and tick “Always load this library on entry” and finally click OK on the dialog box that will appear, as shown in Figure 3.

The plugin is now available. In order to remove it, just remove or delete the “MuscleForceDirection.dll” file from the “plugins” folder.

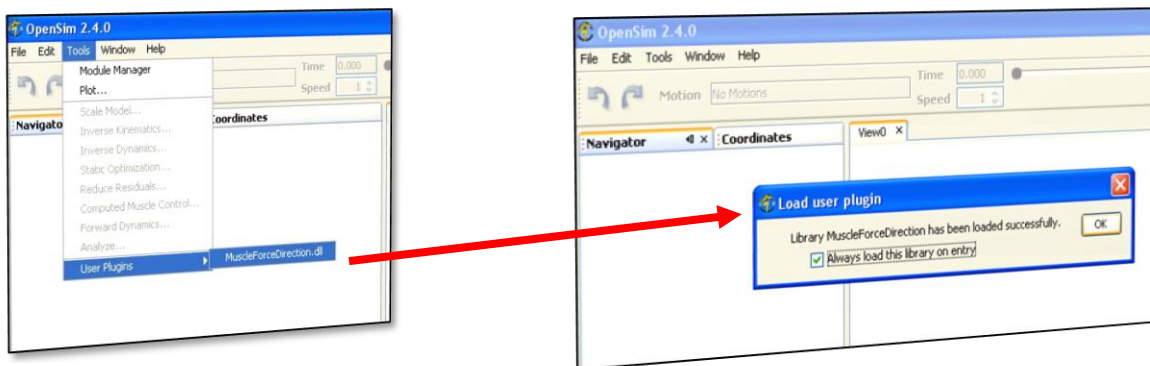


FIGURE 3 STEPS TO FOLLOW IN ORDER TO LOAD THE PLUGIN FROM THE OPENSIM GUI

### 2.2 Inputs and Outputs

The diagram in Figure 4 represents the file involved in the MuscleForceDirection analysis.

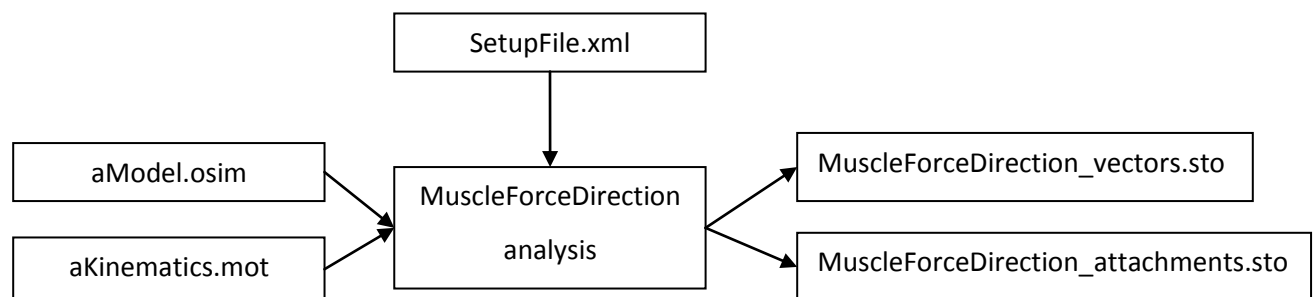


FIGURE 4 DIAGRAM REPRESENTING THE INPUT AND OUT FILES FOR THE MUSCLEFORCEDIRECTION ANALYSIS.

## 2.3 Using the MuscleForceDirection analysis from the GUI

Please copy the “Arm26” folder included in this release to your “Models” directory (typically “C:\OpenSim2.4.0\Models”) and load the model arm26.osim, then follow the operations (also shown in Figure 5 and

Figure 6):

1. From the menu “Tools” choose “Analyze”
2. In the “Main Setting” tab choose “Motion”>”From file” and then select the file “elbow\_flexion.mot” included in this package.

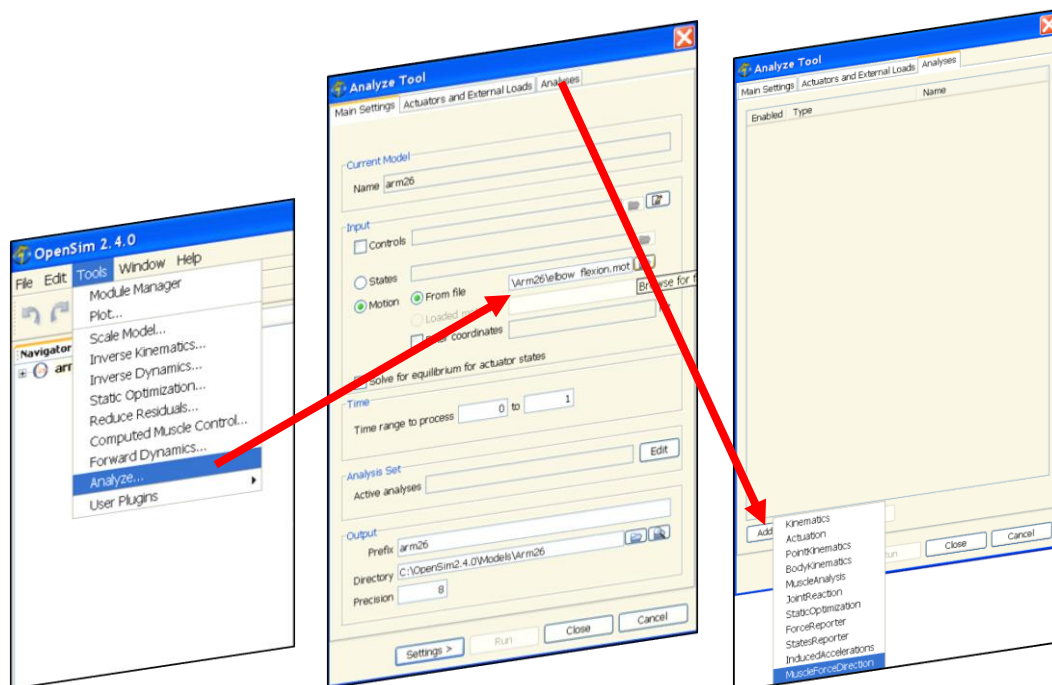


FIGURE 5 SEQUENCE OF OPERATION TO FOLLOW IN ORDER TO SPECIFY THE KINEMATICS AND OPEN THE MUSCLEFORCEDIRECTION PANEL

3. From the “Analyses” tab, press “Add” and select the “MuscleForceDirection” analysis.
4. After adding the MuscleForceDirection analysis click on “Edit”. The analysis panel will show up.
5. Select the desired property values e.g. tick the “print\_attachments” box.
6. Press “OK”, then “Run” in order to start the analysis.

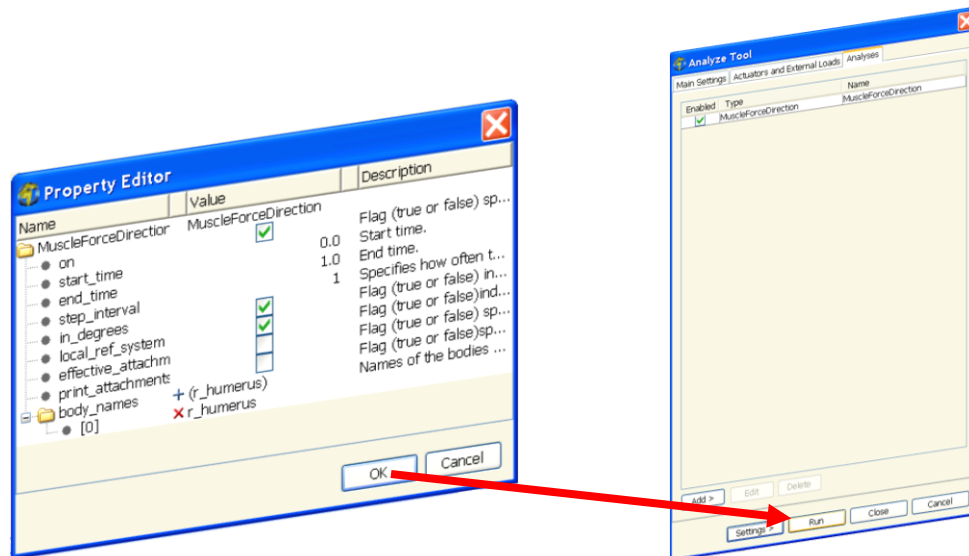


FIGURE 6 HOW TO SET UP THE PROPERTIES AND RUN THE ANALYSIS.

### 2.3.1 The properties in the MuscleForceDirection analysis panel

The Property Editor of the MuscleForceDirection analysis (see Figure 6) presents some common options to all the other analysis (see the OpenSim user guide) and some original options, described in Table 1.

TABLE 1 LIST OF THE PROPERTIES TO BE SET IN THE MUSCLEFORCEDIRECTION ANALYSIS.

Property	Default value	Description
local_ref_system	true	If true, the results of the analysis (both directions and attachments) will be expressed in the local reference system of the segments, specified in the column header of the result file.
effective_attachments	false	If changed to true, effective muscle attachments are considered in the analysis.
print_attachments	false	If changed to true, the muscle attachments where the muscle lines of actions are calculated will be printed.
body_names	'all'	Selection of bodies whose attached muscles will be considered in the analysis. The keyword 'all' executes the analysis on all the bodies (and muscles).

### 2.3.2 Setup file

An XML file to setup the analysis as previously described through the GUI can be seen below, with the XML code shown in red defining the MuscleForceDirection analysis. The possible values taken by the individual fields are the same as described for the GUI in Table 1.

```
<OpenSimDocument Version="20302">
  <AnalyzeTool name="ArmFlexion">
    <model_file> arm26.osim </model_file>
    <results_directory> . </results_directory>
    <output_precision> 12 </output_precision>
    <initial_time> 0.00000000 </initial_time>
    <final_time> 1.00000000 </final_time>
    <AnalysisSet name="Analyses">
      <objects>
        <MuscleForceDirection name="MuscleForceDirection">
          <on> true </on>
          <start_time> 0.00000000 </start_time>
          <end_time> 1.00000000 </end_time>
          <step_interval> 1 </step_interval>
          <in_degrees> true </in_degrees>
          <local_ref_system> true </local_ref_system>
          <effective_attachments> false </effective_attachments>
          <print_attachments> true </print_attachments>
          <body_names> all </body_names>
        </MuscleForceDirection>
      </objects>
      <groups/>
    </AnalysisSet>
    <coordinates_file> elbow_flexion.mot </coordinates_file>
    <lowpass_cutoff_frequency_for_coordinates> 6.0 </lowpass_cutoff_frequency_for_coordinates>
  </AnalyzeTool>
</OpenSimDocument>
```

### 2.4 Using the MuscleForceDirection analysis from the command line

The following command can be used to run the MuscleForceDirection analysis from the command line.

**analyze -S SetupFile.xml -L MuscleForceDirection**

Please note that a copy of the dll has to be in the working directory together with the setup file.



## 2.5 Calling the MuscleForceDirection analysis from Matlab

For less experienced OpenSim users an example of Matlab script that loads the MuscleForceDirection plugin and calls the associated analysis is reported below.

```
clear; clc;

%%%%%%%%%% SETUP THE SIMULATION %%%%%%%%%%%
% current Matlab folder
home_folder = pwd;
% folder where the model and the setup file are.
analysis_folder = 'C:\OpenSim2.4.0\Models\Arm26';
% OpenSim files
plugin_to_load = 'MuscleForceDirection.dll';
SetupFile = 'MySetupFile.xml';
%%%%%%%%%%

cd(analysis_folder);
% run MuscleForceDirection analysis
[message,result]=dos(['analyze -S ',SetupFile,' -L ',plugin_to_load],'-echo');
%importing data
ResultsInMatlab = importdata('ArmFlexion_MuscleForceDirection_vectors.sto');
cd(home_folder)
```

## 2.6 The output files

As represented in Figure 4 the analysis can generate two output files:

- **MuscleForceDirection\_vectors.sto**: this file contains the normalized vectors representing the directions of the muscle lines of action. The vector is always pointing from the selected body where the attachment is located outwards. The body in whose reference system the vector is expressed is always reported as the final part of the column header of each muscle.
- **MuscleForceDirection\_attachments.sto**: this file is optional and prints the coordinates of the muscle attachments. If the user choice is to express the anatomical muscle attachments in the local reference system, the file will contain the first and last muscle points specified for that muscle in the original model file.

For each considered muscle, the files contain results calculated for both the origin and the insertion, to ease muscle directionality analyses not related to finite element modeling. As a

result of this choice, for a model including  $n$  muscles both output files are expected to contain  $6*n+1$  columns, i.e. the time column plus three coordinates or three vector components for each muscle attachment.

## ***2.7 How MuscleForceDirection works***

The MuscleForceDirection plugin executes a few simple operations. Given a selected body (or a set of bodies) included in an OpenSim model, the plugin:

1. identifies the muscles attached to the segment(s).
2. retrieves the current path for each muscle, including wrapping points, by using the `GetPointForceDirections` method of the class `OpenSim::GeometryPath`.
3. Identifies the anatomical or effective muscle attachments according to the user selection and calculates the muscle force direction at that point<sup>1</sup>.
4. if necessary, transforms the previous coordinates and vectors by using the methods of the `OpenSim::SimbodyEngine` class.
5. Prints the muscle force directions and, if requested, the muscle attachments.

## **3 Acknowledgments**

The development of this plugin was started as part of his M.Sc thesis by Alfred Thibon, whose contribution remains substantial and greatly appreciated.

---

<sup>1</sup> As `GetPointForceDirection` does not return the muscle directions associated with wrapping points, the lines of action are calculated simply as difference between the attachment position and the following (previous for the insertion) point of the path in the global reference system.