Automation Of Volumetric Mesh Generation, Mesh Assembly And Model Input From Surface Representations Of Tissue Structures

Ben Landis, **Ahmet Erdemir, PhD**
Department of Biomedical Engineering, Lerner Research Institute, Cleveland Clinic

Motivation

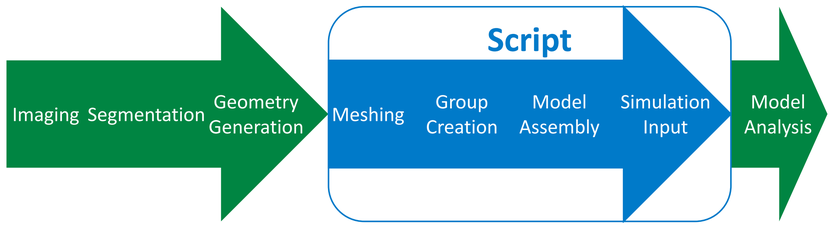
Finite element analysis has become an essential component of biomechanics research. But development of quality finite element representations is laborious process that includes:

- Medical imaging of the region of interest
- Segmentation of the tissues of interest
- Generation of surface representations of tissues
- Automatic volumetric meshing from surface
- Creation of mesh groups for materials, loading, boundary conditions and interactions
- Assembly of individual tissues into model
- Prepare simulation model input file
- Simulate model and analysis

Tools that facilitate high throughput model assembly would reduce this workload significantly and enable large scale simulation in a timely fashion.

Goals

Automate model development processes to assist unsupervised modeling to match complexity and flexibility necessary for biomechanics research.



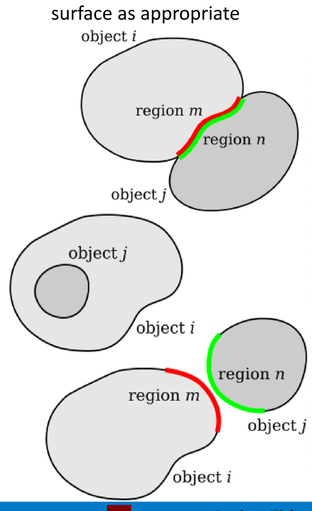
Demonstrate the effectiveness of automation through biomedically relevant case studies.

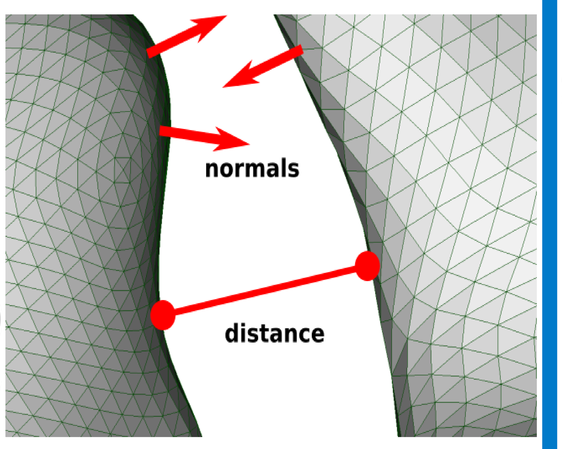
Specifications

A script was developed in Python[1] utilizing the meshing capabilities of Salome[2].

- **Input** - XML file that defines the hierarchy for model assembly and the file containing the surface representations
- **Output** – multiple simulation software formats widely used in biomechanics
- **Constraints** - defined by geometric relationships
  - Ties - fixed displacement between parts
  - Contact - surfaces do not interpenetrate
- **Geometric principles** are used to define mesh regions of these constraints:
  - All – all surfaces used, Contact default
  - Proximity – find all nodes on paired surface within a distance related to the element size within the surface of the first, Ties default
  - Normals – select surface normal vectors that point toward the center of mass of the paired mesh to select region
  - Contains – one part is inside of the other, the region selects the inner or outer surface as appropriate

```
<Assembly>
   <FirstPart>
      <file>First.stl</file>
      <material>rigid</material>
      <Tie>
         <SecondPart multiplier = "0.7"/>
      </Tie>
      <Contact>
         <ThirdPart type="normals"/>
      </Contact>
   </FirstPart>
   <SecondPart>
      <file>Second.stl</file>
      <material>rigid</material>
      <Tie>
         <FirstPart/>
      </Tie>
      <Contact>
      </Contact>
   </SecondPart>
   <ThirdPart>
      <file>Third.stl</file>
      <material>rigid</material>
      <Tie>
      </Tie>
      <Contact>
         <FirstPart type="normal"/>
      </Contact>
   </ThirdPart>
</Assembly>
```



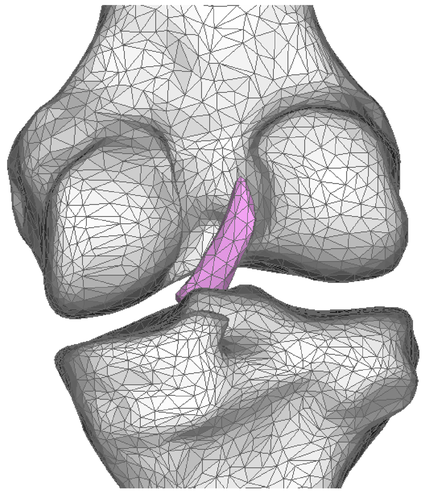


Case Study 1 – Simple Knee

Goals:

- Volumetric meshing of surfaces
- Successfully **produce mesh groups**
- Creation of working simulation model
- Set tie condition between parts

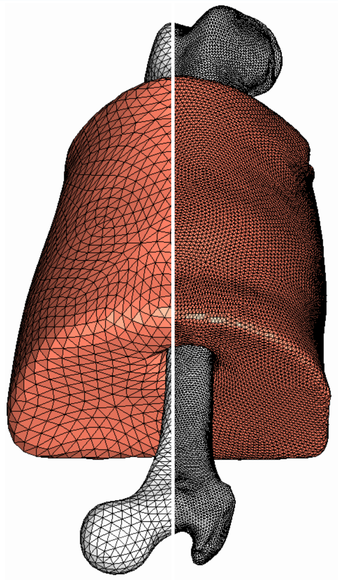
```
<Tibia>
  <file>Tibia.stl</file>
  <material>rigid</material>
  <Contact></Contact>
  <Tie>
    <ACL type="proximity"/>
  </Tie>
</Tibia>
<ACL>
  <file>ACL.stl</file>
  <material>elastic</material>
  <Contact></Contact>
  <Tie>
    <Femur/>
    <Tibia/>
  </Tie>
</ACL>
<Femur>
  <file>Femur.stl</file>
  <material>rigid</material>
  <Contact>
  <Tie>
    <ACL type="proximity"/>
  </Tie>
</Femur>
```



Included in model:

- Femur
- Anterior cruciate ligament
- Tibia

Case Study 2 – Lumped Leg Tissue



Goals:

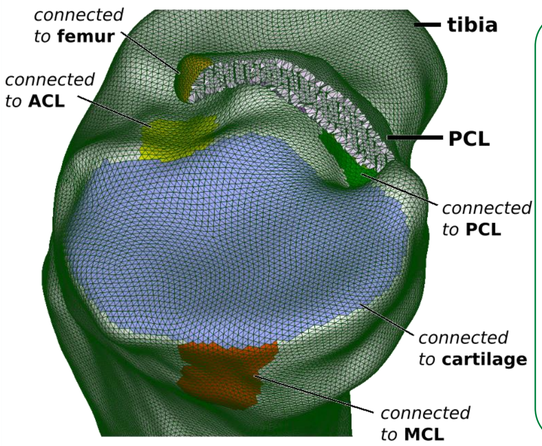
- Apply methodology to different model
- Swap surface representations and rebuild model easily

```
<Tissue>
  <file>Tissue.stl</file>  swap
  <file>Tissue_refined.stl</file>
  <material>elastic</material>
  <Contact></Contact>
  <Tie>
    <Femur type="proximity"/>
  </Tie>
</Tissue>
<Femur>
  <file>Femur.stl</file>  swap
  <file>Femur_refined.stl</file>
  <material>rigid</material>
  <Contact></Contact>
  <Tie>
    <Tissue type="proximity"/>
  </Tie>
</Femur>
```

Case Study 3 – More Detailed Knee

Goals:

- Increase complexity of model
- Include contact in model
- Showcase groups selected by script




```
<Tibia>
  <file>Tibia.stl</file>
  <material>rigid</material>
  <Contact>
    <MCL/>
    <PCL/>
  </Contact>
  <Tie>
    <ACL/>
    <PCL type="proximity"/>
    <MCL/>
    <Cartilage/>
  </Tie>
</Tibia>
<PCL>
  <file>PCL.stl</file>
  <material>elastic</material>
  <Contact></Contact>
  <Tie>
    <Femur/>
    <Tibia/>
  </Tie>
</PCL>
```

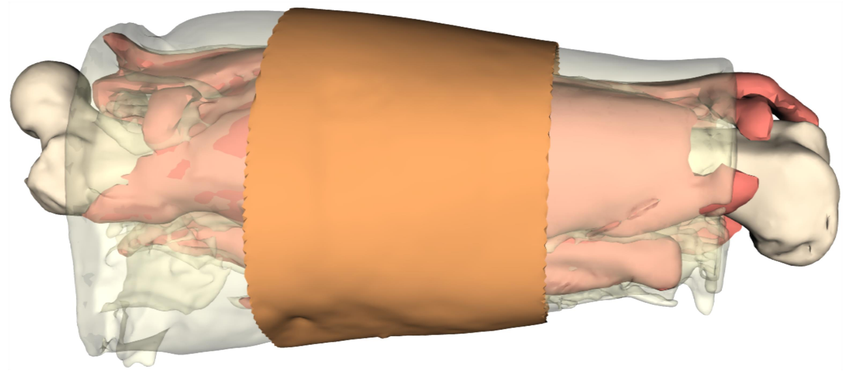
Case Study 4 – Multi-Layered Leg Tissue

Goals

- Add complexity to leg tissue model
- Required the development of the "contains" method to create contact or tie for thin structures that contain other

Included in model:

- Femur
- Muscle
- Fat
- Skin



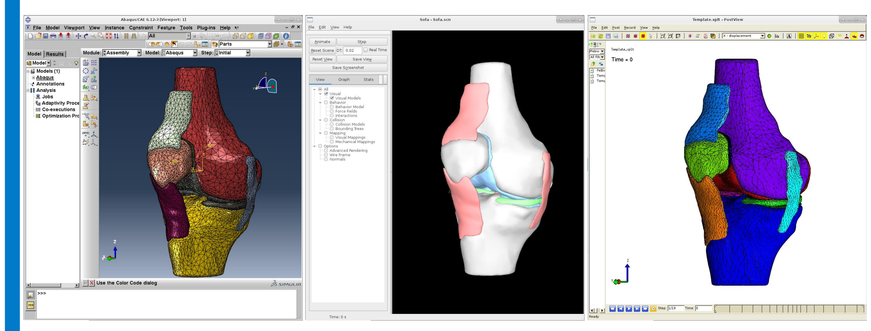
Case Study 5 – Full Knee Model

Goals:

- Introduce the full complexity of modeling in the Open Knee(s) [6] project
- Produce simulation ready model in multiple formats with identical meshes, with all of the handles in place to apply boundary conditions
  - Abaqus [3]
  - FEBio [4]
  - SOFA Framework [5]

Simulation in **preferred** format with **identical mesh**

Abaqus **SOFA Framework** **FEBio**



Conclusions

The presented scripting strategy is a general tool to automate workflow for combining segmented tissue parts for finite element analysis and returning the outcome as the input formats of several simulation software packages. This high throughput approach removes the model assembly and definition bottleneck of finite element analysis. Hundreds of models can be quickly produced from a library of geometries.

At its current state, the tool has some limitations that could be eliminated in the future. Currently all surface representations are assumed to be in the same coordinate system. Swapping a part for diseased or artificial replacement would require registration. Additionally surface representations are required to be triangulated, but could easily be extended to any type supported by Salome.

References

[1] Python https://www.python.org
[2] Salome http://www.salome-platform.org
[3] Abaqus http://www.simulia.com
[4] FEBio https://febio.org
[5] SOFA https://www.sofa-framework.org/
[6] Open Knee(s) https://simtk.org/projects/openknee
[7] Operation Multis https://simtk.org/projects/multis

Project Title: Open Knee(s) – Virtual Biomechanical Representations of the Knee Joint
Project Funding: NIGMS, NIH (R01GM104439; PI Erdemir)
Project Website: https://simtk.org/projects/openknee
Contact: Ahmet Erdemir, erdemir@ccf.org
Contributors: Craig Bennets, Tara Bonner, Snehal Chokhandre, Robb Colbrunn, Scott Delp, Ahmet Erdemir, Omar Gad, Jason Halloran, Henry Kwong, Joy Ku, Ben Landis, Connor Lough, Steve Maas, Raghav Malik, Scott Sibole, Katie Stemmer, Diana Suciu, Jeff Weiss

Project Title: Reference Models for Multi-Layer Thickness Structures
Project Funding: USAMRMC, DoD (W81XWH-15-1-0232; PI Erdemir)
Project Website: https://simtl.org/projects/multis
Contact: Ahmet Erdemir, erdemir@ccf.org
Contributors: Tara Bonner, Robb Colbrunn, Scott Delp, Ahmet Erdemir, John Eric Jelovsek, Joy Ku, Ben Landis, Erica Morrill, Tammy Owings, Tyler Schimmoeller, Mike Wong