A Prototype for Relational Assembly and Templating of Multi-Component Models of Biological Structures for Finite Element Analysis

Benjamin Landis, Ahmet Erdemir PhD

Department of Biomedical Engineering, Cleveland Clinic, Cleveland, USA

Provide full correspondence details here including e-mail for the corresponding author

Provide short biographical notes on all contributors here if the journal requires them.

This is using the author template for the journal

Computer Methods in Biomechanics and Biomedical Engineering.

A Prototype for Relational Assembly and Templating of Multi-Component Models of Biological Structures for Finite Element Analysis

Computational modelling reduces the need for in vivo and in vitro experimentation and the necessity of physical prototyping of interventions. However the development of virtual representations for finite element analysis is laborious. Mesh generation has existing methods for automation. An underappreciated aspect of development is the definition of mesh locations where interactions can be prescribed. These are commonly done interactively through a user interface in pre-processing software. An unsupervised strategy to automatically define mesh regions based on relational anatomy will lead into high-throughput model assembly. A review of the literature fails to find automated model assembly even in specific fields. This study is to develop a scripted, unsupervised method as a prototype for assembly, templating, and exchange of multi-component finite element representations of biomechanical structures.

Keywords: word; another word; lower case except names; between 3 and 6

Subject classification codes: include these here if the journal requires them

Word Count: 4433

# Introduction

Computational modelling of biological structures, in particular finite element analyses of organs, joints, and tissue are ubiquitous, examples include the eye [1], brain [2] [3], spine [4], knee [5], shoulder [6], dentistry [7], and even prostheses [8] [9]. The popularity of finite element analysis is understandable as computational modelling provides the means to reduce the need for in vivo and/or in vitro experimentation and the necessity of physical prototyping of interventions. Additionally, simulation-based approaches are increasingly leveraged for personalized medicine [10] [11].

The development of quality virtual representations for finite element analysis is a laborious task. Generation of a working model involves multiple steps: (1) segmentation of medical images to define tissue volumes, (2) generation of geometric representations of tissue structures, i.e. triangulated surface, (3) development of finite element meshes of tissue structure, (4) definitions of node, element, face sets to assist in prescription of material properties, loading and boundary conditions, and interactions between different tissue components, (5) assignment of material properties, loading and boundary conditions and interactions, and (6) troubleshooting of model convergence by iterative simulations. Many studies usually do not describe these steps in elaborate detail, i.e. simply reporting what a model is (not necessarily how it is generated) necessitates reporting of approximately 80 parameters. A fundamental bottleneck in the model development workflow is the segmentation of tissues. This is a well-recognized problem and there are many emerging strategies to automate, or at least facilitate, this step [12] [13] [14]. The burden of mesh generation has also been noticed and a large number of methods exist for automation. An underappreciated aspect of model development is the annotation of the mesh to define mesh regions where properties and model inputs can be assigned to and locations at which interactions, e.g. ties and contact between tissue components, can be prescribed. These are commonly done interactively through a graphical user interface in pre-processing software of simulation packages. When geometry of multiple tissue components already exist, a fully unsupervised strategy to generate meshes and to automatically define mesh regions based on relational anatomy will lead into high-throughput model assembly and model templating. A review of the literature fails to demonstrate an automated way to assemble model even in specific fields.

The need for anatomical representation, i.e. tissue meshes including region definitions, is common to all physics-based simulation software. An important consideration for readily assembled models, which can serve as templates for further development, is the possibility to push them in to formats by different simulation tools. The choice of finite element analysis software in the biomechanics community is fragmented. Yet, commonly used simulation platforms include FEBio[15], a free and open source software, and Abaqus [16], a commercial software [17] [18]. For surgical simulations and medical training, specialized software such as SOFA framework [19] are broadly utilized. Choice of software is commonly informed by availability, modeler’s experience, and cost. With increasing multi-institutional nature of modeling studies and the push for sharing and reuse of models and model components [20], exchange of models in different formats becomes more pressing.

The goal of this study is to develop a scripted, unsupervised method as a prototype for assembly, templating, and exchange of multi-component finite element representations of biomechanical structures. Additional aims include the demonstration of this prototype’s capacity to generate different variety of assemblies and models from existing geometric representations and to export template models in multiple simulation software formats.

# Methods

## Unsupervised Model Assembly, Templating, and Export

### Overall Approach

The development of model generating scripts requires all models that will be developed to meet specified criteria that can then be assumed. The first assumption is that the simulation will be mechanical in nature (as opposed to thermal, electrical or otherwise). A second assumption is that the simulation will involve the interaction of two or more parts that already have been placed into the same coordinate space. It will therefore be essential that the user define these parts and types of interactions between them.

### Scripting Environment

The unsupervised group creation has been developed within Salomé 8.3.0 [11] a free and open source software for 3D modeling and mesh generation with built in Python 2.7.10 [12] scripting.

### Inputs

The specification of these simulation parts is provided to the script in the form of an XML file, making the ordering of the parts inclusion irrelevant. The tissues, i.e. the individual parts, need to be explicitly referenced and their surface geometries given. Additionally any constraints, such as a tie or contact between tissues should be specified. This will allow for the automated creation of regions of interest that will be used to apply these constraints in the final finite element model. An optional choice allows the final mesh to be of second order and should be specified if desired.

### Features and Processes

Every constraint has a type that tells the script how to calculate the mesh region. The surface geometries for each part are loaded and the constraints, as defined in the mark-up file, are examined one by one. The parts involved are numerically analyzed node by node to determine where the constraint applies. When all of the applicable nodes have been found then a group is created that will be used when the simulation model is created. These created groups can be both groups of nodes or groups of faces containing those nodes.

The constraints are defined by geometric principles relating the two tissues. The currently defined principles are:

* Proximity – find all nodes on the paired surface within a distance related to the size of the element size. This is the default for Tie constraints.

<ACL type="proximity"/>

* Normals – select faces that have normal vectors that point toward the barycenter of the other tissue. This can also be limited by the proximity if desired.

<LCL type="normals"/>

* Contains – one mesh contains the other, select faces that have normal vectors that point inward or outward.

<MCL type="contains"/>

* All – all the surfaces. This is the default for contact.

<PCL type="all"/>

Two additional groups can be defined. These option only affect an individual mesh:

* Vector – Make a group of the mesh that has normal vectors approximately equal to input vector
* Remaining – a group of all faces not in any other group, this will be create an empty group if all faces are already in groups

After all the constraint groups have been created, then each part's finite element mesh is created. The surface geometries are converted to a native format (MED) without remeshing. These surfaces are then meshed in 3D, unless flagged as rigid. It is this point in the process that meshes can be turned into second order, if desired. These finite element ready meshes are then saved to be used when creating the simulation template.

### Outputs

The automated generation of groups produces and saves intermediary finite element meshes of the parts of the simulation as given in the XML document. Additionally the simulation template containing the all of the part and connections is produced in the format specified. Currently the script can produce out in three formats: Febio v2.5 [10], the Sofa Framework v16.12 [13], and Abaqus v6.2 [14].

# Case Studies

## Study 1. Simplified Knee - Femur, Tibia

### Highlighted Feature/Utility

This study demonstrates the scripts ability to create contact between two parts. It is necessary to test this basic functionality in a simple manageable test case. The resulting simulation template can be simulated to demonstrate the presence of contact.

### Inputs

The inputs required for this study are stls taken from Open Knee. These files were readily available for download. Additionally script’s input connectivity XML had to be generated.

### Outputs

The expected output is a template simulation file. To ensure the contact has been formulated properly formed, this template will be manually adjusted to move the femur towards the tibia. If contact has been applied by the script then a contact force should be present, otherwise these models will simply pass through on another.

## Study 2. Simplified Knee - Femur, ACL, Tibia

### Highlighted Feature/Utility

This study demonstrates the scripts ability to create ties between parts. It is necessary to test this basic functionality in a simple manageable test case. The ACL attached to the femur and tibia was chosen the ties could be easily tested by pulling the bones apart the ligament stretching between them would ensure the tied condition.

### Inputs

The inputs required for this study are stls taken from Open Knee. These files were readily available for download. Additionally script’s input connectivity XML had to be generated.

### Outputs

The expected output is a template simulation file. To ensure the ties are properly formed, this template will be manually adjusted to increase the separation between the femur and tibia enough to see that the ligament is forced to stretch to maintain the ties on both ends.

## Study 3. Simplified Knee - Femur, Femoral Cartilage, Tibial Cartilage, Tibia

### Highlighted Feature/Utility

This study demonstrates the scripts ability to create contact and ties, while still remaining a simple manageable test case. The cartilages will be tied to the respective bones and contact between the cartilages will be set.

### Inputs

The inputs required for this study are stls taken from Open Knee. These files were readily available for download. Additionally script’s input connectivity XML had to be generated.

### Outputs

The expected output is a template simulation file. Similarly to Study 1, this template could be manually adjusted to move the femur towards the tibia. The ties should hold the cartilage to the bones, while a contact force should be found as the cartilages are pressed into contact.

## Study 4. Full Knee

### Highlighted Feature/Utility

This study demonstrates the full capability and convenience of an automated workflow. A knee model with sixteen separate parts and a total of thirty one connections, manually creating these groups and setting the boundary conditions is tedious prone to error. The script can produce the model effortlessly.

### Inputs

The inputs required for this study are stls taken from Open Knee. These files were readily available for download. Additionally script’s input connectivity XML had to be generated.

### Outputs

The expected output is a template simulation file. The simulation is complex and testing is beyond the scope of this study.

## Study 5. Lumped Soft Tissue and Bone with Ultrasound Probe

### Highlighted Feature/Utility

This study demonstrates the ability to use parts that are not biological in nature. A femur with a fleshy part attached with a nearby ultrasound probe which will be set to contact the flesh.

### Inputs

The inputs required for this study are stls taken from Operation Multis. These files were readily available for download. Additionally script’s input connectivity XML had to be generated.

### Outputs

The expected output is a template simulation file.

## Study 6. Layered Soft Tissue and Bone with Ultrasound Probe

### Highlighted Feature/Utility

This study demonstrates the ability to substitute out one part for another or in this case one part for three layered parts. In Study 5, flesh and bone are attached, here the flesh is replaced with layers of muscle, fat, and skin. Each layer will be in contact with the adjacent layers. The ultrasound probe will be set for contact with the skin in the study.

### Inputs

The inputs required for this study are stls taken from Operation Multis. These files were readily available for download. Additionally script’s input connectivity XML had to be generated.

### Outputs

The expected output is a template simulation file.

# Results

## Prototype Scripts

A prepared dissemination package is available at place holder until we create a downloadable package where we disseminate at the project site. MAKE A LINK

## Study 1. Simplified Knee - Femur, Tibia

The resulting template is simulated demonstrating that contact was properly formed, as shown in Figure *2*.

A prepared dissemination package including all input and output files, including model files and any simulation results is available at placeholder. MAKE A LINK

## Study 2. Simplified Knee - Femur, ACL, Tibia

The ACL is tied to the femur and tibia, the knee is rotated by one radian. A linear elastic material is used for the ACL. Figure 3 shows the stress produced while the joint moves and the ties hold.

A prepared dissemination package including all input and output files, including model files and any simulation results is available at placeholder. MAKE A LINK

## Study 3. Simplified Knee - Femur, Femoral Cartilage, Tibial Cartilage, Tibia

The femur was moved under displacement control into the tibia. Contact occurs between the femoral cartilage and the medial and lateral tibial cartilage, Figure 4 shows this pressure.

A prepared dissemination package including all input and output files, including model files and any simulation results is available at placeholder. MAKE A LINK

## Study 4. Full Knee

This study of the full knee successful shows the true utility of the script as assigning the groups by hand would be tedious and time consuming but the computer handles it effortlessly. This simulation is complex and beyond the scope of this study, however the full knee was loaded into each of the simulation platforms, see Figure *5*, and simulation start was verified.

A prepared dissemination package including all input and output files, including model files and any simulation results is available at placeholder. MAKE A LINK

## Study 5. Lumped Soft Tissue and Bone with Ultrasound Probe

This study demonstrates that parts that are non-anatomical can be added to the simulation template.

A prepared dissemination package including all input and output files, including model files and any simulation results is available at placeholder. MAKE A LINK

## Study 6. Layered Soft Tissue and Bone with Ultrasound Probe

This study demonstrates that one part can be removed with a different geometry in its place. In particular this study remove the lumped flesh of the leg and replaces it with the muscle fat and skin as individual layers a s seen in Figure 6.

A prepared dissemination package including all input and output files, including model files and any simulation results is available at placeholder. MAKE A LINK

## [AE NOTES ON DISCUSSION →

* It seems like the organization of the Discussion is in following. This is fine but needs to be elaborated more:
  + Summary and broad implications
  + Determination of regions of interaction (automated/unsupervised annotation)
  + Output to different simulation software (exchange)
  + Full knee example (also implying the need for automation)
  + Limitations and potential future studies
  + Take home message.
* A more in-depth look at the importance of each feature in relevance to case studies is imperative. There are hints of this but it is rather unorganized and studies to generate the referred figures do not seem to be described in the methods section well. Why are these important, how it is done otherwise? Remember the motivation in the introduction and emphasize how now the missing need is addressed.
* How does the method compare to interactive strategies to define mesh. A more detail step by step example of how this would have been done in the traditional way will be helpful. The full knee example hints this but does not explicitly state how the assembly and template model is generated interactively, e.g. what are the steps.
* Discussion is a good place to synthesize the studies contributions in relation to existing literature.
  + A review of the literature fail to demonstrate an automated way to assemble models even in specific fields.

# Discussion

This paper presented the development of a scripted, unsupervised method to create a prototype for assembly, templating, and exchange of multi-component finite element representations of biomechanical structures. The capacity to generate a variety of assemblies and models was demonstrated allowing existing geometric representations to be rapidly processed in a high throughput approach removes the model assembly bottleneck of finite element analysis, allowing a large quantity of models can be quickly produced from a library of geometries which can then be exported to template models in multiple simulation software formats.

When geometry of multiple tissue components already exist, a fully unsupervised strategy to generate meshes and to automatically define mesh regions based on relational anatomy will lead into high-throughput model assembly and model templating.

The need for anatomical representation, i.e. tissue meshes including region definitions, is common to all physics-based simulation software. An important consideration for readily assembled models, which can serve as templates for further development, is the possibility to push them in to formats by different simulation tools. The choice of finite element analysis software in the biomechanics community is fragmented. Yet, commonly used simulation platforms include FEBio[15], a free and open source software, and Abaqus [16], a commercial software [17] [18]. For surgical simulations and medical training, specialized software such as SOFA framework [19] are broadly utilized. Choice of software is commonly informed by availability, modeler’s experience, and cost. With increasing multi-institutional nature of modeling studies and the push for sharing and reuse of models and model components [20], exchange of models in different formats becomes more pressing.

An important part of the process is the automatic creation of the subsets of the meshes that are used to assign boundary conditions. Figure 7 shows an example of this applied to the knee. While this automation is a necessary feature, it is also necessary that the user can fine tune the resulting group(s). The proximity or normals methods for creating groups can be enlarged or reduced by adding an xml attribute multiplier. This multiplier increases the area of interest within the mesh as seen in Figure 8.

Once the regions of interest have been found the script will create a model template for simulation. This template contains all of the constraints defined in the markup file and will have placeholders where a simulation user can apply boundary condition as desired. With little effort the template can be turned into a working model and simulations begun.

Before launching the script, definition of the process needs to be defined in the form of an XML file where the individual parts are referenced and constraints asserted. Upon running the script the surface geometries for each part are loaded and the constraints, as defined in the XML, are numerically examined one by one and a group is created. After the constraints then part is made into a finite element mesh. These finite element meshes are then used to create the simulation model template.

The utility of the script greatly increases with increased usability. Therefore it is important for the script to produce a simulation ready template model in the desired format. Because each of the simulation formats is produced from the same mesh, the result will be identical to the element and node, and applied boundary conditions, and material properties. The current version of the script produces input file for Febio, Abaqus, and SOFA Framework simulations.

The full knee model that was developed is also a good example of the benefits of automating the model assembly process, with 16 separate parts and a total of 31 connections between them, manually creating each of these groups and then setting the boundary conditions is tedious, time consuming, and prone to user error. Figure 9 shows the groups on the tibia made by the script. Figure 5 shows the full knee model was produced and loaded into each simulation environment.

Scripting the model generation process allows for high throughput and removes the model assembly bottleneck of finite element analysis. A large volume of models can be quickly produced from a library of geometries, and mesh convergence studies and can be rapidly generated by swapping component geometries with higher or lower resolutions. This frees the user from the tedium of model generation and allows time to be spent elsewhere.

As a scripted process some limitations do exist. The models produced are limited to solid mechanics simulations. The geometries themselves are assumed to be stl files, a common but not ubiquitous file format. The process itself relies on access to a library of tissue geometries, and further makes the presumption that the tissues are the same scale and in the same coordinate system. This is true for geometries generated from the same set of medical images, but must be verified by the user if pieces do not come from the same source. And for volumetric meshing, these surface geometries must also be watertight. Even then the resulting meshes will be tetrahedral, while many groups may prefer hexahedral elements. Finally a key limitations of automated group creation is the possibility of creating groups that are inappropriate to the model, but are not noticed due to an inattentive user. The knee model provides a good example of this. If either meniscus is tied by proximity but the multiplier is large, the whole of meniscus could be tied, and effectively function as a rigid body if the groups is not reviewed.

Additionally some mechanical models may require the definition of mesh groups that cannot be produced with the current set of geometric components. Also the expansion of the model generation possibilities to include additional of physics, such as thermal modeling, could prove to be a useful extension. These extensions could be added if the need arises.

# Acknowledgments

Operation MULTIS or “Reference Models for Multi-Layer Tissue Structures" was made possible by a contract vehicle which was awarded and administered by the U.S. Army Medical Research & Materiel Command under award number: W81XWH-15-1-0232.

Open Knee(s) is currently funded by the National Institute of General Medical Sciences, National Institutes of Health (1R01GM104139). Previous activities were partially funded by the National Institute of Biomedical Imaging and Bioengineering, National Institutes of Health (1R01EB009643).

# Availability

All of the scripts and models can be found at <https://simtk.org/projects/multis>.

# References

[1] T. M. Nejad, C. Foster, D. Gongal, T. M. Nejad, C. Foster, and D. Gongal, “Finite element modelling of cornea mechanics: a review,” *Arq. Bras. Oftalmol.*, vol. 77, no. 1, pp. 60–65, Feb. 2014.

[2] J. S. Giudice, W. Zeng, T. Wu, A. Alshareef, D. F. Shedd, and M. B. Panzer, “An Analytical Review of the Numerical Methods used for Finite Element Modeling of Traumatic Brain Injury,” *Ann. Biomed. Eng.*, Oct. 2018.

[3] M. I. Miga, “Computational Modeling for Enhancing Soft Tissue Image Guided Surgery: An Application in Neurosurgery,” *Ann. Biomed. Eng.*, vol. 44, no. 1, pp. 128–138, Jan. 2016.

[4] H. Schmidt, F. Galbusera, A. Rohlmann, and A. Shirazi-Adl, “What have we learned from finite element model studies of lumbar intervertebral discs in the past four decades?,” *J. Biomech.*, vol. 46, no. 14, pp. 2342–2355, Sep. 2013.

[5] F. Galbusera *et al.*, “Material Models and Properties in the Finite Element Analysis of Knee Ligaments: A Literature Review,” *Front. Bioeng. Biotechnol.*, vol. 2, Nov. 2014.

[6] M. Zheng, Z. Zou, P. jorge D. silva Bartolo, C. Peach, and L. Ren, “Finite element models of the human shoulder complex: a review of their clinical implications and modelling techniques,” *Int. J. Numer. Methods Biomed. Eng.*, vol. 33, no. 2, p. e02777, Feb. 2017.

[7] N. Murakami and N. Wakabayashi, “Finite element contact analysis as a critical technique in dental biomechanics: A review,” *J. Prosthodont. Res.*, vol. 58, no. 2, pp. 92–101, Apr. 2014.

[8] Z. H. Zhou *et al.*, “Improved anchoring nails: design and analysis of resistance ability,” *BMC Oral Health*, vol. 18, Aug. 2018.

[9] J. Maminskas, A. Puisys, R. Kuoppala, A. Raustia, and G. Juodzbalys, “The Prosthetic Influence and Biomechanics on Peri-Implant Strain: a Systematic Literature Review of Finite Element Studies,” *J. Oral Maxillofac. Res.*, vol. 7, no. 3, Sep. 2016.

[10] “Welcome to Python.org,” *Python.org*. [Online]. Available: https://www.python.org/. [Accessed: 20-Dec-2018].

[11] “Welcome to the www.salome-platform.org — SALOME Platform.” [Online]. Available: https://www.salome-platform.org/. [Accessed: 10-Dec-2018].

[12] “FEBio Software Suite.” [Online]. Available: https://febio.org/. [Accessed: 20-Dec-2018].

[13] “SOFA.” [Online]. Available: https://www.sofa-framework.org/. [Accessed: 20-Dec-2018].

[14] “Abaqus Unified FEA - SIMULIATM by Dassault Systèmes®.” [Online]. Available: https://www.3ds.com/products-services/simulia/products/abaqus/. [Accessed: 20-Dec-2018].

Table 1: Full connectivity of the Knee Model. C denotes contact between parts, T denotes a tie. Parts may be tied with contact.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Femur | Tibia | Fibula | Patella | Femoral Cartilage | Tibial  Lateral Cartilage | Tibial  Medial Cartilage | Patellar Cartilage | Medial Meniscus | Lateral Meniscus | ACL | PCL | MCL | LCL | Patellar Ligament | Quadriceps Tendon |
| Femur | -- |  |  |  | T |  |  |  |  |  | T | T | TC | TC |  | C |
| Tibia |  | -- |  |  |  | T | T |  | T | T | T | T | TC |  | T |  |
| Fibula |  |  | -- |  |  |  |  |  |  |  |  |  |  | T |  |  |
| Patella |  |  |  | -- |  |  |  |  |  |  |  |  |  |  | T | T |
| Femoral Cartilage | T |  |  |  | -- | C | C | C | C | C |  |  |  |  |  | C |
| Tibial Lateral Cartilage |  | T |  |  | C | -- |  |  |  | C |  |  |  |  |  |  |
| Tibial Medial Cartilage |  | T |  |  | C |  | -- |  | C |  |  |  |  |  |  |  |
| Patellar Cartilage |  |  |  |  | C |  |  | -- |  |  |  |  |  |  |  |  |
| Medial Meniscus |  | T |  |  | C |  | C |  | -- |  |  |  | T |  |  |  |
| Lateral Meniscus |  | T |  |  | C | C |  |  |  | -- |  |  |  |  |  |  |
| ACL | T | T |  |  |  |  |  |  |  |  | -- | C |  |  |  |  |
| PCL | T | T |  |  |  |  |  |  |  |  | C | -- |  |  |  |  |
| MCL | TC | TC |  |  |  |  |  | T |  |  |  |  | -- |  |  |  |
| LCL | TC |  | T |  |  |  |  |  |  |  |  |  |  | -- |  |  |
| Patellar Ligament |  | T |  | T |  |  |  |  |  |  |  |  |  |  | -- |  |
| Quadriceps Tendon | C |  |  | T | C |  |  |  |  |  |  |  |  |  |  | -- |

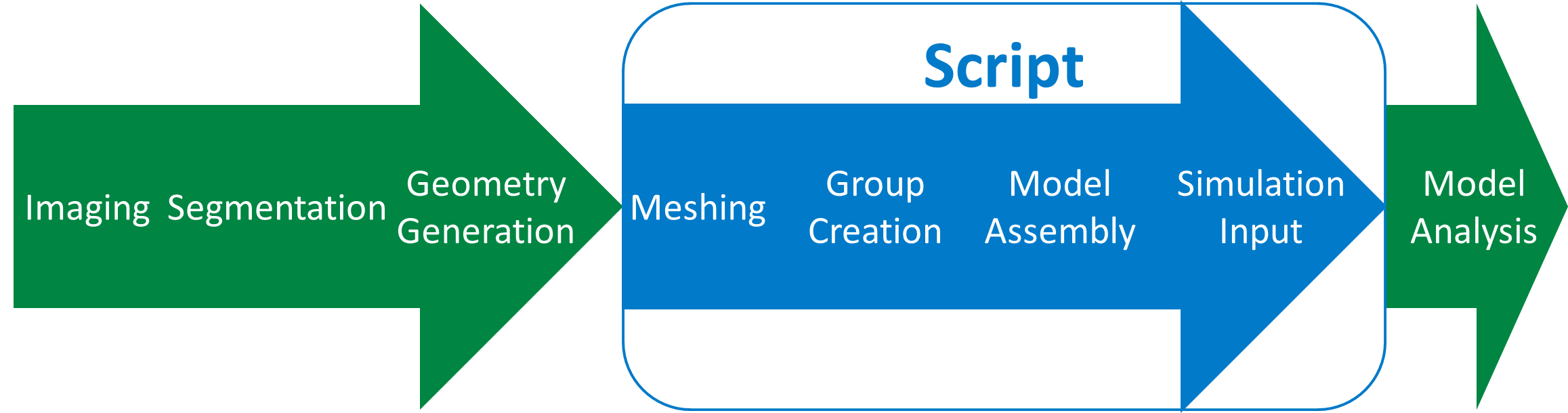


Figure 1: Describing the data flow with a figure.

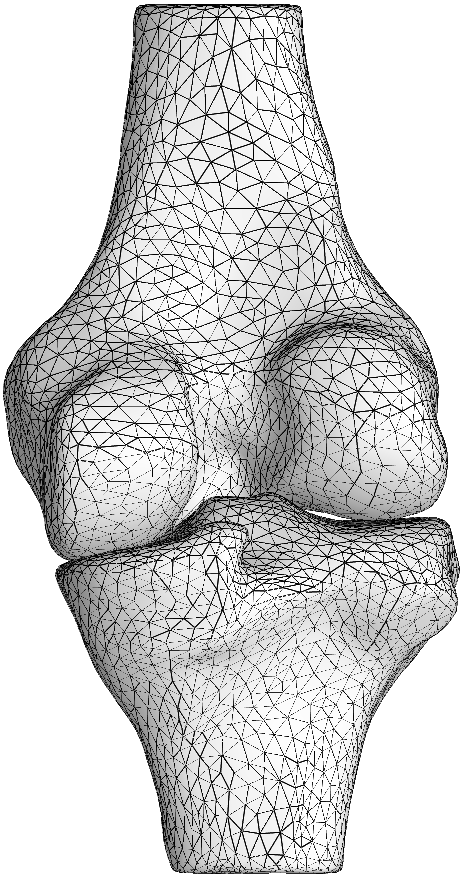


Figure 2: Contact between Femur and Tibia



Figure 3: The ties hold and generate a stress in the ACL while bending.

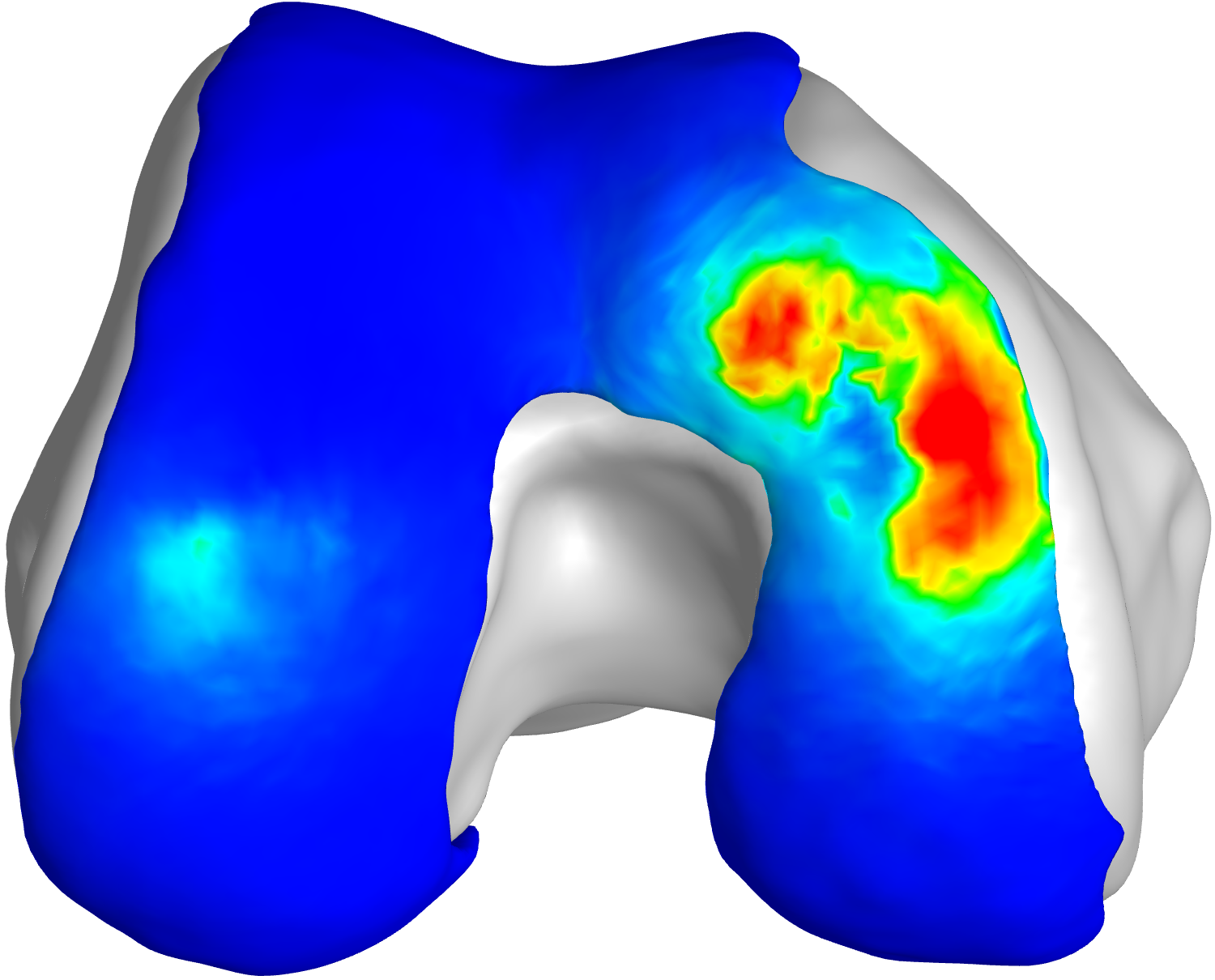


Figure 4: Stress on the femoral cartilage caused by contact with tibial cartilage.

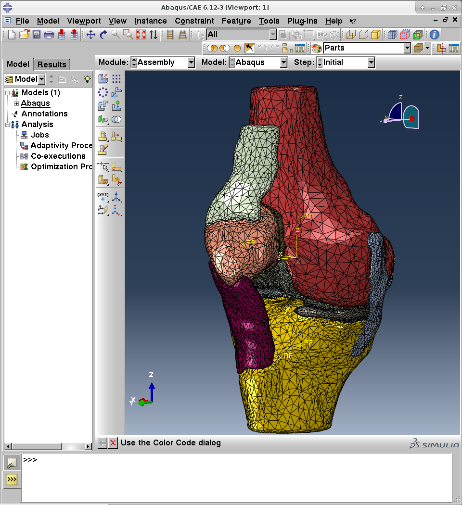
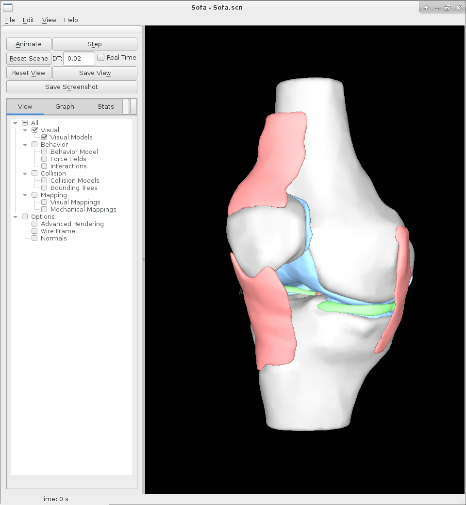
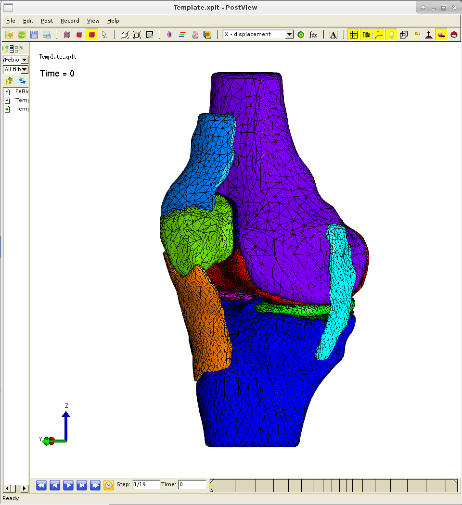


Figure 5: The same full knee model in different simulation software packages. Left to right, Febio, the Sofa Framework, and Abaqus.

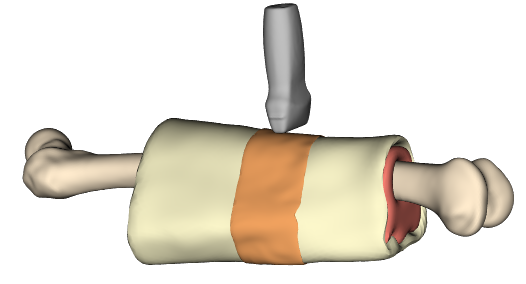


Figure 6: Layered Leg

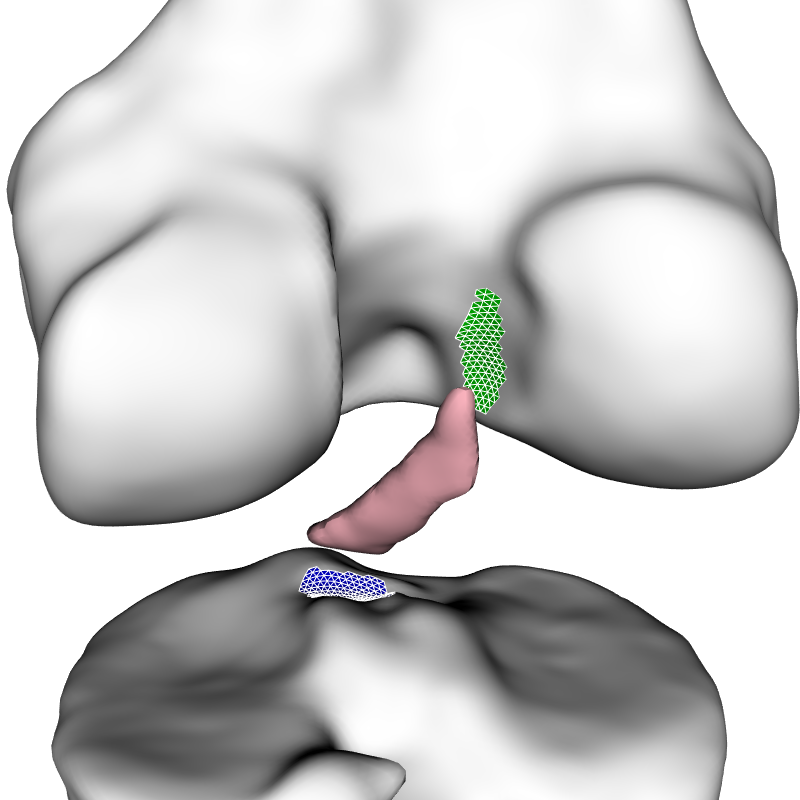


Figure 7: The femur and tibia have been slightly moved away from the ACL so that the groups that will be used to create ties can be seen.

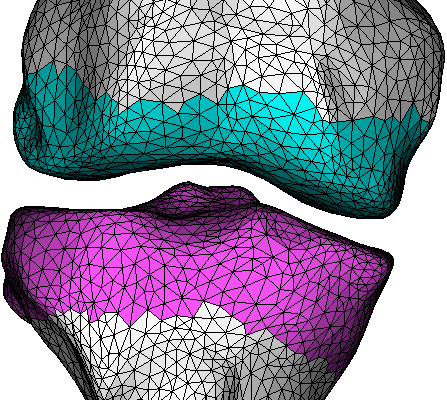
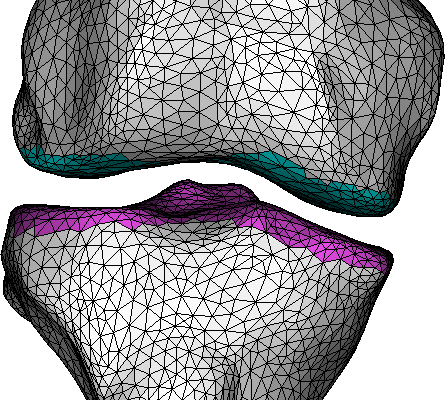


Figure 8: the proximity modifier allows for the automated group size to be altered.

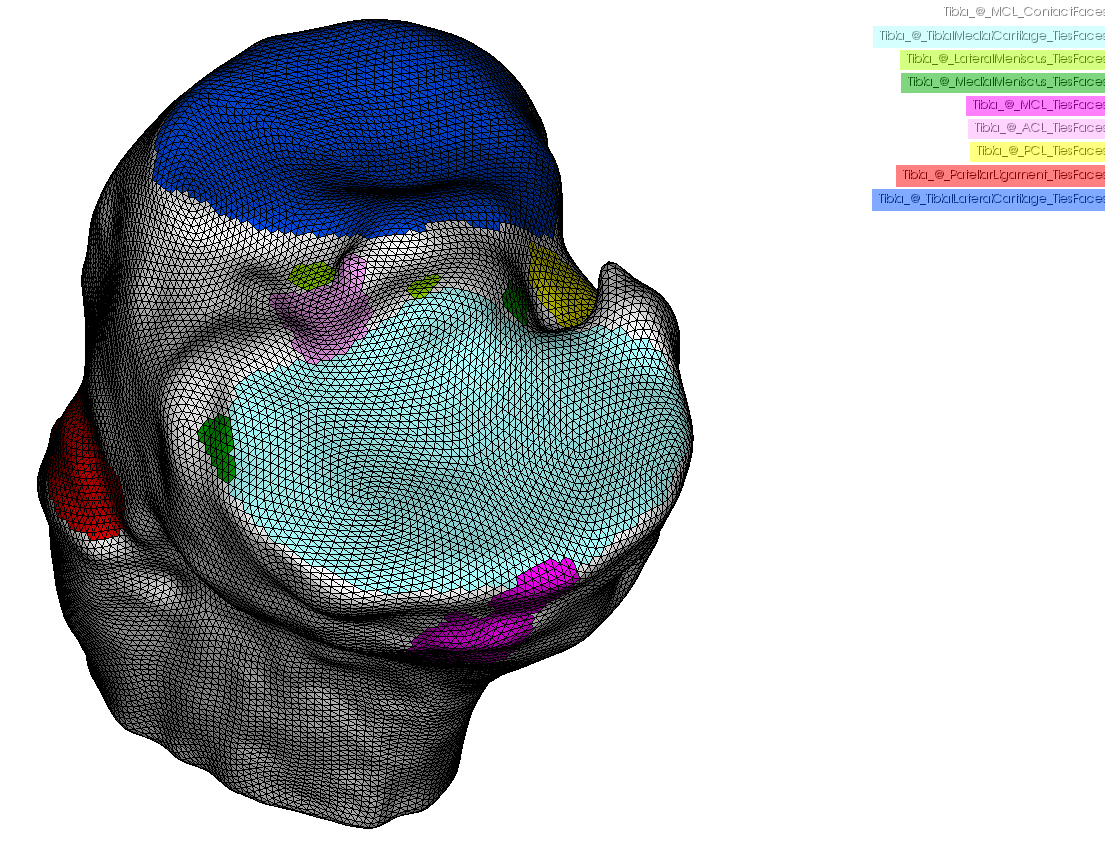


Figure 9: Groups on the tibia as generated by the scripted process.