gruner

Geotechnical Simulations with Moose

– DRAFT –

This whitepaper collects information and code
for geotechnical simulations with the software Moose.

Jörg Meier

Gruner AG, Geotechnical Engineering
Basel, Switzerland

10.10.2024

Contents

Chapter 1

Introduction

The open-source, parallel finite element framework Moose (mooseframework.inl.gov) can also be used to perform geotechnical simulations. This document aims to collect options and techniques that are suitable for creating such geotechnical simulations with mosses. Main focus are quasi-static simulations.

This document is primarily intended as a collection of information rather than a textbook or step-by-step guide.

Chapter 2

Geometry

2.1 General Considerations

When modelling geotechnical problems, it is often desirable to make a meaningful geometric abstraction to avoid time-consuming modelling, but also to avoid inaccuracies due to oversimplification. However, experience shows that the effort involved in creating such a model geometry should not be underestimated and that the models consist of a large number of individual parts and components - often called 'geometrical objects'. Some of these objects have trivial shapes (e.g. rectangular surfaces for a sheet pile wall segment), others can have non-trivial curvilinear boundaries (e.g. geological bodies). All of these objects must be shaped to fit each other (without gaps or overlaps) and together form the FE model. Usually, geotechnical models are managed (e.g materials, staging, etc) on the level of these objects.

2.2 'Subdomains' resp. 'Blocks'

Geometrically, a Moose model consists of one or more 'subdomains'. Due to the use of libmesh by Moose, the term 'block' is also used as a synonym for 'subdomain' in many places. Element types and Moose objects such as materials, kernels, etc. are assigned to these subdomains in turn. Furthermore, each FE element belongs to exactly one subdomain.

It is therefore necessary to create at least one subdomain for each element type and each material. To simplify the handling of construction stages, it has also proved useful to further subdivide element sets into individual subdomains so they match the 'geometrical objects' mentioned before so they can to be used independently during the simulation (e.g. deactivated or activated at different times). This document follows this approach that Moose models are managed on the basis of subdomains/blocks.

As the shape of these individual parts and components directly influences the shape of the FE elements they are build of, further prerequisites must be considered during geometrisation: For instance, as few 'unfavourably shaped' (e.g. too acute-angled) FE elements as possible should result in order to avoid numerical instabilities or load increments that are too small. A common reason for acute-angled elements and large numbers of elements are corner or end points that are close together and unfavourable intersections of the objects. Another requirement for geometrisation is that the number of FE elements should remain within a range that enables acceptable calculation times.

2.3 Getting the mesh into Moose

Although Moose's 'Mesh System' provides a variety of tools for generating FE meshes on the basis of input file commands, the creation of geometrically complex models with these commands is often confusing and error-prone.

Therefore, the mesh including all subdomains and FE-elements is prepared outside of Moose and imported using the `FileMeshGenerator`. The MSH file format has proven itself in this context. One possible workflow is to create the geometry of the subdomains in Blender, then generate the mesh using gmsdh and save it as an MSH file. See listing 2.1 for the code required in a Moose input file for the import of an MSH file.

Listing 2.1: Read mesh from a file

```
[Mesh]
    [file]
        type = FileMeshGenerator
        file = "source.msh"
    []
    second_order = true
[]
```

LowerDimensionalBlocks in MSH-files

Chapter 3

Modelling of geometric entities, structures, loads, etc.

This chapter collects information on how different types of geometric entities, structures, loads etc. can be modelled.

(to be written)

3.1 Volumes

3.2 FixedEndSprings

https://mooseframework.inl.gov/source/nodalkernels/CoupledForceNodalKernel.html

https://onlinelibrary.wiley.com/doi/pdf/10.1002/pamm.202200045

3.3 Springs between nodes (NodeToNodeAnchors)

https://mooseframework.inl.gov/mastodon/source/materials/LinearSpring.html

3.4 Beams (line element)

3.5 EmbeddedBeams (line element)

3.6 Plates (surface element)

3.7 Interfaces (surface element)

3.8 Contact

3.9 Pore Pressure Boundaries

Chapter 4

Material Models

(to be written)

Chapter 5

General Model Setup

(to be written)

Calculation type:

- Incremental Small Strains Strains.html
- Stress Stresses.html

Chapter 6

Initial Conditions

(to be written)

- Initial Stress State

Chapter 7

Construction Stages

(to be written)

- Construction Stages
a. Activating and Deactivating Elements
b. Replacing Elements
c. Adjusting Prestress

Chapter 8

Postprocessing

(to be written)