

基于 Taichi 语言的高性能离散元 数值模拟平台

TaichiDEM V1.0

使
用
说
明
书

Multiscale Geomechanics Lab@ZJU

目录

| | |
|-------------------|----|
| 1. 引言..... | 3 |
| 1.1 编写目的..... | 3 |
| 1.2 研发背景..... | 3 |
| 2. 软件概述..... | 4 |
| 2.1 运行环境..... | 4 |
| 2.2 软件介绍..... | 4 |
| 2.2 软件安装..... | 5 |
| 2.3 软件结构..... | 7 |
| 2.4 模型参数输入介绍..... | 8 |
| 3. 案例及运行结果..... | 27 |
| 3.1 三轴排水试验..... | 27 |
| 3.2 自然休止角..... | 29 |

1. 引言

1.1 编写目的

本设计说明书详细地描述了“高性能离散元数值模拟平台 TaichiDEM”的整体框架，软件功能，以及各个模块的功能和函数使用方法。为了方便岩土工程师以及研究岩土方向的科研人员使用，本文对平台的编译环境和平台使用都进行了系统地说明，在最后结合案例介绍了平台配套的后处理软件的使用方法。

1.2 研发背景

颗粒材料广泛地存在于自然界或工业生产中，其是由大量离散固体颗粒组成的复杂非线性系统。颗粒间的相互作用在宏观上体现为颗粒物质独特的物理力学性质固体或流体的特殊力学特性，在一定条件下发生类固-液转化现象，从而使得整体的宏观物理力学特性更为复杂多变。颗粒材料作为人们为满足一定的生产目的而需要利用或处理的颗粒物质，因此研究颗粒材料的剪切强度和变形机理对工业和工程实践具有重要的指导意义。颗粒材料的多尺度特性对计算效率提出了较高的要求，因此利用基于图像处理器（GPU）的并行算法开发基于球形/多球颗粒的离散元高性能计算平台，满足大规模工程应用的计算要求。

国家教育部于 2018 年提出了《教育部关于开展国家虚拟仿真实验教学项目建设工作的通知》，进一步发挥国家虚拟仿真实验教学项目示范带动作用，助推信息技术与实验教育教学深度融合，助推高等教育教学质量变轨超车。此外十四五报告明确提出，加强核心自主工业软件研发，加快建立核心工业软件备份系统，充分发挥科研院所、企业等各方力量，提炼核心技术难点以及行业关键问题，组织实施核心自主工业软件关键技术攻关工程，协同攻关。

为解决大规模颗粒的数值计算问题以及响应国家高校数字化教学的号召以及十四五规划方案，设计完成了高性能离散元数值模拟平台 TaichiDEM。

2. 软件概述

高性能离散元数值模拟平台 TaichiDEM, 以解决工程实际问题和培养岩土类人才为目标, 以土体试验数据为基础, 工程实际为对象构建数值仿真平台及实验教学体系, 以计算机硬件和软件技术特别是多维多媒体技术为基础, 以现代虚拟仿真技术为手段, 开发面向土体参数实验和工程事故研究的虚拟仿真平台, 全面提高受训人员的岩土基础知识和编程能力。

同时, 该平台为岩土类工程师在工程设计和施工中提供一种计算依据, 能有效地提高实际工程的可靠度, 可为国家基础设施建设做出重要的贡献。

2.1 运行环境

本平台可以在多核 CPU 或 GPU 架构以及 Windows/Linux/Mac 系统上运行。

2.2 软件介绍

使用 Taichi 语言设计并实现了一种岩土工程问题数值计算的高性能计算平台 TaichiDEM (Taichi-based Discrete Element Method for geophysical problems)。TaichiDEM 主要实现了百万级别以上的大规模颗粒模拟, 该代码已成功植入多场多尺度高性能求解器 GeoTaichi 中。

平台的整体组成和主要的消息传递关系如图 1 所示, 程序整体组成主要分为基础层、核心层、计算层以及用户层四个部分。

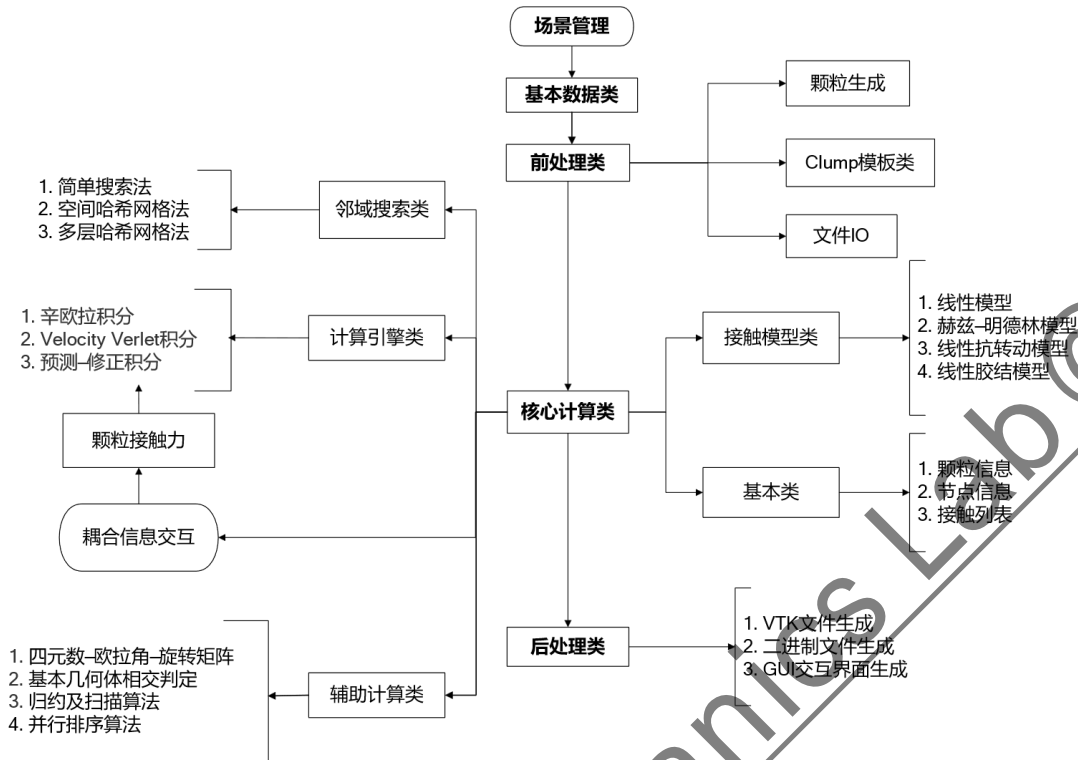


图 1 软件设计流程图

1. 基础层包含离散元程序计算所需的基本信息，包括颗粒信息（位置、速度、角速度、质量、转动惯量等），边界信息（节点位置及面片法向）、接触信息（颗粒ID、接触力、历史切向力）以及材料信息（密度、阻尼）等。
2. 核心层包括包含了整个程序中所有需要在 GPU 上计算的核函数，实现了前后处理、时间积分循环、时间步内层循环、收敛性判断等功能，其包括邻域搜索类、接触类型类、计算引擎类、辅助计算类、前处理类、后处理类等。
3. 在计算层将这些核函数根据特定的计算需要进行组装，包含了消息传递的主要路径。最终基于 Python 的面向对象功能，构造各类计算功能函数的 API 供用户调用。
4. 在用户层中，用户可以根据具体工程调用预设 API，进行数值模拟。

2.2 软件安装

平台语言采用 Taichi 并行语言进行编写，并了调用部分 Python 库，电脑环境需要满足以下要求：

- (1) 操作系统需要 Windows 10/Windows11 或者 Ubuntu18/Ubuntu20/Ubuntu22。

(2) 硬件需要 1.8GHZ 以上的处理器，内存至少需要 2GB RAM，硬盘空间需要 800M 以上可用空间。

(3) 若需要开启 GPU 加速计算功能，则 GPU 显存至少需要 4GB。

(4) Python 版本应满足以下要求：Python \geq 3.8，依赖的 Python 库需要满足以下版本要求：

| | | |
|----------|----|--------|
| pip | == | 23.3.1 |
| pybind11 | == | 2.10.4 |
| taichi | == | 1.6.0 |
| imageio | == | 2.25.1 |
| numpy | == | 1.23.5 |
| trimesh | == | 3.20.1 |
| cmake | == | 3.18.2 |
| eigen | == | 3.4.0 |

在本手册中，仅针对 Ubuntu 系统的安装做详细说明，对于 Windows 及 Mac 系统的安装与 Ubuntu 系统类似：

(1) 进入安装文件夹并下载软件：

```
cd /path/to/desired/location/  
git clone https://github.com/Yihao-Shi/GeoTaichi  
cd GeoTaichi
```

(2) 安装必要的依赖库

```
# 安装 Python 与 pip  
sudo apt-get install python3.8 python3-pip  
# 安装 Python 依赖包  
python3 -m pip install taichi imageio pybind11 numpy trimesh psutil -i  
https://pypi.douban.com/simple  
# 安装 C++ 中 Eigen 库  
sudo apt-get install libeigen3-dev cmake==3.18.2  
# 安装 pybind11  
cd third_party  
git clone https://github.com/pybind/pybind11.git  
cd pybind11  
mkdir build  
cmake ..  
make check -j 4  
sudo make install
```

(3) 若需要进行 GPU 加速计算，需要在 Ubuntu 系统上安装 Cuda Toolkit，具体安装步骤详见 <https://docs.nvidia.com/cuda/cuda-installation-guide-linux/index.html>

(4) 设置环境变量

```
sudo gedit ~/.bashrc  
$ export PATH="$PATH:/path/to/desired/location/GeoTaichi"  
source ~/.bashrc
```

(5) 编译 C++编写的部分代码

```
python src/utis/NURBS/setup.py install --user
```

2.3 软件结构

本设计基于严谨的理论推导，结合大量的土体参数试验数据，使用 Taichi 语言设计并实现了一种岩土工程问题数值计算平台 TaichiDEM。平台一共包括了：基础层、核心层、计算层以及用户层四个部分。每个部分中的模块采用面向对象的思维进行编写，并做到模块之间的函数可以重复使用计算结果不会相互影响。平台的运行顺序为：

- (1) 读取计算工况参数，并初始化；
- (2) 判断颗粒最大位置是否超过阈值，若未超过则跳至步骤（4）；
- (3) 将颗粒投影至预先生成的空间哈希网格中，形成内存连续的颗粒—网格列表；
- (4) 进行宽接触搜索并建立潜在接触列表，在此基础上压缩潜在接触列表，形成接触列表；
- (5) 进行窄接触搜索，并利用接触本构模型计算接触力/力矩，并将其加至每个颗粒上；
- (6) 计算每个颗粒的体力与阻尼；
- (7) 根据不同的积分方式进行时间积分，更新颗粒的位置与速度；
- (8) 重置颗粒的接触力/力矩，进入新的时间步循环。

具体计算流程如图 2 所示。

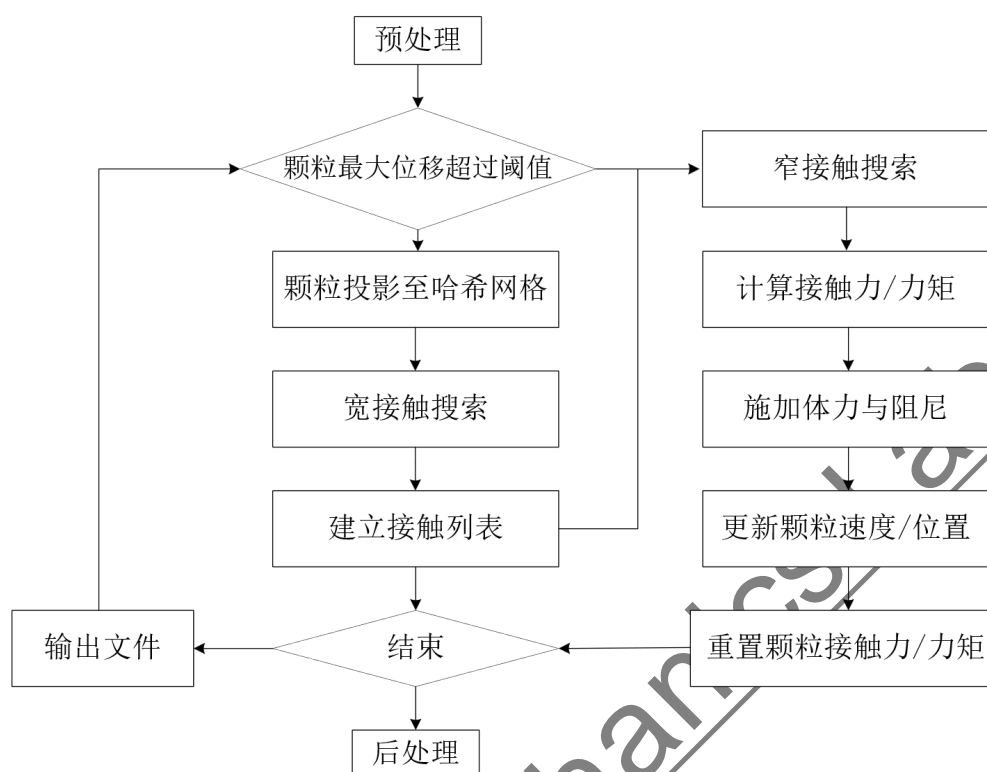


图 2 DEM 计算流程

2.4 模型参数输入介绍

TaichiDEM 采用 Python 字典作为输入创建离散元信息, 计算平台通过字典中的关键字读取其附带的信息, 其中关键字不分先后顺序。

(1) 初始化

| | |
|-------------------------------------|---|
| init(keywords ...) | |
| Primary keywords: | |
| ① arch [str] [option] | 选择运行后端, 可选为“cpu”或者“gpu”, 默认值为“gpu”。 |
| ② cpu_max_num_thread [int] [option] | 若运行后端为 CPU, 可指定最大运行线程数, 默认值为该 CPU 最大的线程数。 |
| ③ offline_cache [bool] [option] | 选择是否储存编译后的文件, 默认值为 True。 |

| | |
|---|--|
| ④ debug [bool] [option] | 选择是否开启 debug 模式，默认值为 False。 |
| ⑤ default_fp [str] [option] | 选择浮点数计算精度，可选为“float64”或“float32”，默认为“float64”。 |
| ⑥ default_ip [str] [option] | 选择整型计算精度，可选为“int32”或“int64”，默认为“int32”。 |
| ⑦ device_memory_GB [float] [option] | 若计算后端为 GPU，预先分配需要计算的内存，默认为 2GB。 |
| ⑧ device_memory_fraction [float] [option] | 若计算后端为 GPU，预先分配需要计算的内存，默认为 80%。 |
| ⑨ kernel_profiler [bool] [option] | 选择是否统计核函数的运行时间，默认值为 False。 |

(2) 基本布局参数

| | |
|---------------------------------|---|
| set_configuration(keywords ...) | |
| Primary keywords: | |
| ① domain [list] | 设定计算区域大小。 |
| ② boundary [list] [option] | 设定计算边界类型，可选为“destroy”、“reflect”或“period”，默认值为[“destroy”, “destroy”, “destroy”]。 <u>注意“period”功能尚未开发。</u> |
| ③ gravity [list] [option] | 设定场地重力，默认值为[0., 0., -9.8]。 |
| ④ engine [str] [option] | 选择求解器类型，可选为“SymplecticEuler”、“VelocityVerlet”或“PredictCorrector”，默认值为“SymplecticEuler”。 |
| ④ search [str] [option] | 选择邻居搜索算法，可选为“BrustSearch”、“LinkedCell”或“HierarchicalLinkedCell”，默认值为“LinkedCell”。 <u>注意“HierarchicalLinkedCell”功能尚未完全开发。</u> |

(3) 内存分配参数

| | | |
|-------------------------------|--|--|
| memory_allocate(keywords ...) | | |
| Primary keywords: | | |
| ① memory [dict] | | |
| | max_material_number [int] | 最大材料个数。 |
| | max_particle_number [int] | 最大颗粒个数。 |
| | max_sphere_number [int] [option] | 最大球体类型个数，默认值为 0。 |
| | max_clump_number [int] [option] | 最大多球类型个数，默认值为 0。 |
| | max_plane_number [int] [option] (墙体类型为 plane) | 最大平面个数，模拟中所有墙体均为无限大平面，默认值为 0，需要注意无法与“max_facet_number”以及“max_patch_number”共同设置。 |
| | max_facet_number [int] [option] (墙体类型为 facet) | 最大三角墙体个数，默认值为 0，模拟中多边形墙体将自动被离散为三角墙体，需要注意无法与“max_plane_number”以及“max_patch_number”共同设置。 |
| | max_patch_number [int] [option] (墙体类型为 patch) | 最大三角面片个数，默认值为 0，模拟中复杂边界均被三角面片离散，需要注意的是无法与“max_plane_number”以及“max_facet_number”共同设置。 |
| | max_servo_wall_number [int] [option] | 最大伺服墙个数，默认值为 0，其中“max_facet_number”必须大于 0。 |
| | compaction_ratio [int] [option] | 接触列表的压缩率，用于优化计算内存占用，默认值为 0.5。 |
| | body_coordination_number [int] [option] | 颗粒最大配位数，默认值为 16。 |

| | | |
|-----------------------|---|--|
| | wall_coordination_number [int] [option] | 颗粒周围最大墙体数，默认值为预先设置的最大墙体个数。 |
| | verlet_distance_multiplier [int] [option] | Verlet 系数,用于建立潜在接触列表，默认值为 0.2。 |
| | wall_per_cell [int] [option] | 当墙体类型为plane 或者 facet 时，设置每个哈希网格中最多的墙体个数，默认为 4。 |
| ② log [bool] [option] | | 是否显示关键的设置参数，可选为 True 或 False，默认值为 True。 |

(4) 求解器参数

| | | |
|--------------------------|----------------------------------|---------------------------|
| set_solver(keywords ...) | | |
| Primary keywords: | | |
| ① solver [dict] | | |
| | Timestep [float] | 初始时间步长 |
| | SimulationTime [float] | 总模拟时间 |
| | CFL [float] [option] | CFL 系数，默认值为 0.5。 |
| | AdaptiveTimestep [bool] [option] | 是否开启自适应时间步，默认值为 False。 |
| | SaveInterval [float] | 存储模拟数据的间隔时间。 |
| | SavePath [str] [option] | 存储数据路径，默认为“./OutputData”。 |

(5) 基本区域参数

| | | |
|--------------------------|------------|---|
| add_region(keywords ...) | | |
| Primary keywords: | | |
| ① region [dict / list] | | |
| | Type [str] | 设定区域形状，可选为“Rectangle”、“TrianglarPrism”、“Spheroid”、“Cylinder”或“UserDefined”。 |
| | Name [str] | 设定区域名字。 |

| | | |
|--|---|--|
| | zdirection [list] [option] | 该区域特征方向向量，默认为 [0., 0., 1.]。 |
| | BoundingBoxPoint [list] | 该区域包围盒的左下角坐标。 |
| | BoundingBoxSize [list] | 该区域的范围。 |
| | Region_volume [taichi.function] [option] | 若区域类型为“UserDefined”，需要定义一类型为 taichi.function 的函数，用作该区域体积的计算，默认值为 None。 |
| | RegionFunction [taichi.function] [option] | 若区域类型为“UserDefined”，需要定义一类型为 taichi.function 的函数，用作判断坐标是否落入用户定义的区域，其中该函数需要两个传入参数，分别为 (position, radius=0)，其中 position 代表一坐标，radius 代表该坐标对应的特征半径，默认值为 None。 |
| | RotateCenter [list] [option] | 定义区域的旋转中心，配合关键词“zdirection”完成区域的旋转，默认值为该区域包围盒的中心。 |

(6) 多球模板参数

| | | | |
|----------------------------------|------------------------------|-----------------|----------------|
| add_clump_template(keywords ...) | | | |
| Primary keywords: | | | |
| ① template [dict / list] | | | |
| | Name [str] | | 设置多球模板的名字。 |
| | NSphere [int] | | 设置多球模板中子球的数量。 |
| | Pebble | Position [list] | 设置每个子球的空间坐标。 |
| | [dict / list] | Radius [float] | 设置每个子球的半径。 |
| | ClumpProperty [str] [option] | | 设置计算多球模板本质属性的算 |

| | | | |
|--|-------------------------------|----------------------|--|
| | | | 法，可选为“MonteCarlo”或“Grid”方法，默认为“Grid”。 |
| | Resolution [float] [option] | | 设置计算多球模板本质属性的精度，当计算方法为“Grid”时生效，默认认为 80（例：背景网格的尺寸为最小子球半径的 1/80）。 |
| | TryNumber [int] [option] | | 设置计算多球模板本质属性的精度，当计算方法为“MonteCarlo”时生效，默认认为 200000。 |
| | ClumpProperty [dict] [option] | Volume [float] | 由用户直接指定多球模板的体积。 |
| | | InertiaMoment [list] | 由用户直接制定多球模板的惯性张量，多球模板的初始密度为 1。 |

(7) 基本颗粒参数

a) 颗粒 create 模板

| | | | |
|---------------------------|------------------------|--|------------------------------|
| create_body(keywords ...) | | | |
| Primary keywords: | | | |
| ① body [dict / list] | | | |
| | BodyType [str] | | 设定颗粒的类型，可选为“Sphere”或“Clump”。 |
| | Template [dict / list] | Radius [float] | 设定颗粒的半径。 |
| | | BodyPoint [list] | 设定颗粒的坐标。 |
| | | GroupID [int] | 设定颗粒所属组的 ID。 |
| | | MaterialID [int] | 设定颗粒所属材料的 ID。 |
| | | InitialVelocity [list] [option] | 设定颗粒的初速度，默认值为[0., 0., 0.]。 |
| | | InitialAngularVelocity [list] [option] | 设定颗粒的初始角速度，默认值为[0., 0., 0.]。 |
| | | FixVelocity [list] [option] | 设定颗粒的平动自由 |

| | | | |
|--|--|--------------------------------------|---|
| | | | 度，默认值为[0., 0., 0.]。 |
| | | FixAngularVelocity [list] [option] | 设定颗粒的转动自由度，默认值为[0., 0., 0.]。 |
| | | BodyOrientation [list] [option] | 设定生成颗粒朝向的概率分布类型，在此仅可选为“constant”或 None，默认值为 None。 |
| | | OrientationParameter [list] [option] | 设定颗粒的朝向，当颗粒类型为“Clump”时生效，默认值为[0., 0., 1.]。 |

b) 颗粒 generate / distribute 模板

| | | |
|------------------------|------------------------------|---|
| add_body(keywords ...) | | |
| Primary keywords: | | |
| ① body [dict / list] | | |
| | BodyType [str] | 设定颗粒集合体的类型，可选为 “Sphere” 或 “Clump”。 |
| | GenerateType [str] | 设定颗粒集合体的生成类型，可选为“Generate”或“Distribute”。 |
| | RegionName [str] | 设定颗粒集合体生成所在区域的名字。 |
| | Period [list] [option] | 设定颗粒集合体生成的时间间隔，输入格式为[开始时间, 结束时间, 时间间隔]，默认值为[0, 0, 1e6]。 |
| | CheckHistory [bool] [option] | 设定颗粒在生成时是否需要避免与已生成颗粒 |

| | | | |
|--|---------------------------------|----------------------------|--|
| | | | 的重叠，默认为 True。 |
| | WriteFile [bool] [option] | | 设定颗粒生成后是否需要将生成信息以 txt 格式输出，默认为 False。 |
| | Visualize [bool] [option] | | 设定颗粒生成后是否输出可视化文件，默认为 False。 |
| | TryNumber [int] [option] | | 设定颗粒生成时尝试次数，默认为 100。 |
| | PossionSampling [bool] [option] | | 设定颗粒生成是否采用泊松采样算法，当颗粒集合生成方式为“Generate”时生效，默认为 False。 |
| | Porosity [int] [option] | | 设定颗粒集合体的初始孔隙比，当颗粒集合生成方式为“Distribute”时生效，默认是为 0.345。 |
| | Template [dict / list] | Radius [float] | 设定颗粒的半径（若颗粒集合中所有颗粒半径均相同，可用关键词“Radius”代替“MaxRadius”和“MinRadius”）。 |
| | | MaxRadius (Radius) [float] | 设定颗粒集合的最小半径。 |
| | | MinRadius (Radius) [float] | 设定颗粒集合的最大半径。 |
| | | BodyNumber [int] | 设定颗粒集合体所需生成的颗粒数量，当颗粒集 |

| | | | |
|--|--|--|---|
| | | | 合生成方式为“Generate”时生效。 |
| | | Fraction [float] [option] | 设定颗粒集合体某次生成颗粒的百分比，当颗粒集合生成方式为“Distribute”时生效，默认值为 1。 |
| | | GroupID [int] | 设定颗粒所属组的 ID。 |
| | | MaterialID [int] | 设定颗粒所属材料的 ID。 |
| | | InitialVelocity [list] [option] | 设定颗粒的初速度，默认值为[0., 0., 0.]。 |
| | | InitialAngularVelocity [list] [option] | 设定颗粒的初始角速度，默认值为[0., 0., 0.]。 |
| | | FixVelocity [list] [option] | 设定颗粒的平动自由度，默认值为[0., 0., 0.]。 |
| | | FixAngularVelocity [list] [option] | 设定颗粒的转动自由度，默认值为[0., 0., 0.]。 |
| | | BodyOrientation [str] [option] | 设定生成颗粒朝向的概率分布类型，可选为“constant”、“uniform”、“gaussian”或 None，默认值为 None。 |
| | | OrientationParameter [list] [option] | 设定颗粒的朝向，当颗粒类型为“Clump”时生效，默认值为[0., 0., 1.]。 |

c) 颗粒文件读取模板

| |
|---------------------------------|
| add_body_from_file(keyword ...) |
| Primary keywords: |

| ① Body [dict / list] | | |
|----------------------|------------------------|---|
| | FileType [str] | 设定需要读取文件的类型，可选为“TXT”、“NPZ”或“OBJ”。 |
| | BodyType [str] | 设定生成颗粒的类型，可选为“Sphere”或“Clump”，当读取文件类型为“TXT”时生效。 |
| | Period [list] [option] | 设定颗粒集合体生成的时间间隔，输入格式为[开始时间，结束时间，时间间隔]，默认值为[0, 0, 1e6]。 |
| | Template [dict / list] | ParticleFile [str] [option] |
| | | ClumpFile [str] [option] |
| | | PebbleFile [str] [option] |
| | | GeometryFile [str] |
| | | GroupID [int] |
| | | MaterialID [int] |

| | | | |
|--|--|--|--|
| | | InitialVelocity [list] [option] | 设定颗粒的初速度，默认值为[0., 0., 0.]。 |
| | | InitialAngularVelocity [list] [option] | 设定颗粒的初始角速度，默认值为[0., 0., 0.]。 |
| | | FixVelocity [list] [option] | 设定颗粒的平动自由度，默认值为[0., 0., 0.]。 |
| | | FixAngularVelocity [list] [option] | 设定颗粒的转动自由度，默认值为[0., 0., 0.]。 |
| | | ScaleFactor [int] [option] | 设定体素的放大系数， 当文件类型为“OBJ”时生效，默认值为 1。 |
| | | Translation [int] [option] | 设定体素的平移距离， 当文件类型为“OBJ”时生效，默认值为[0., 0., 0.]。 |
| | | Orientation [list] [option] | 设定体素的朝向， 当文件类型为“OBJ”时生效，默认值为[0., 0., 1.]。 |

（8）基本墙体参数

a) 墙体生成模板

| | | |
|------------------------|----------------|--|
| add_wall(keywords ...) | | |
| Primary keywords: | | |
| ① wall [dict] | | |
| | WallType [str] | 设定墙体类型，可选为“Plane”、“Facet”或“Patch”。 “Plane”为空间中无限大的平板； “Facet”为空间中有限大的三角平板，可组成空间中多边形墙； “Patch”为空间中小三角形面片，可用于复杂 |

| | | |
|--|---------------------------------|--|
| | | 边界建模。 |
| | WallID [int] | 设定墙体的 ID。 |
| | MaterialID [int] | 设定墙体所属材料的 ID。 |
| | WallCenter [list] | 设定墙体的中心坐标，当墙体类型为“Plane”时生效。 |
| | WallVertex [dict] | 设定多边形墙体的顶点坐标，当墙体类型为“Facet”时生效。 |
| | OuterNormal [list] | 设定墙体的外法线方向，即会发生潜在接触的墙面朝向。 |
| | InitialVelocity [list] [option] | 设定墙体初始速度，当墙体类型为“Facet”时生效，默认值为[0., 0., 0.]。 |
| | ControlType [str] [option] | 设定伺服类型，将墙体转换为伺服墙体，仅可选择“StressControl”，当墙体类型为“Facet”时生效，同时需在函数/memory_allocation/中设置关键词“max_servo_wall_number”，若无需将墙体类型转换为伺服墙，则可采用默认值 None。 |
| | Alpha [float] [option] | 设定墙体伺服的控制参数，当墙体伺服类型为“StressControl”时生效，默认值为 0.8。 |
| | TargetStress [list] | 设定墙体伺服的目标压力，当墙体伺服类型为“StressControl”时生效。 |
| | LimitVelocity [list] | 设定墙体最快移动速度，当墙体伺服类型为“StressControl”时生效。 |
| | WallFile [str] | 设定墙体文件读取路径，当墙体类型为“Patch”时生效。 |
| | ScaleFactor [int] [option] | 设定墙体模型的放大系数，当墙体类型为“Patch”时生效，默认值为 1。 |
| | Translation [list] [option] | 设定墙体模型的平移距离，当墙体类型为“Patch”时生效，默认值为[0., 0., 0.]。 |

| | | |
|--|----------------------------|--|
| | Orientation [int] [option] | 设定墙体模型的朝向，当墙体类型为“Patch”时生效，默认值为[0., 0., 1.]。 |
| | Visualize [bool] [option] | 设定是否输出墙体可视化文件，当墙体类型为“Facet”或“Patch”时生效。默认值为 False。 |

b) 墙体文件读取模板

| | | |
|------------------------|---------------------------|--|
| add_wall(keywords ...) | | |
| Primary keywords: | | |
| ① wall [dict] | | |
| | WallFile [str] | 设定墙体文件读取路径。 |
| | ServoFile [str] [option] | 设定伺服墙体文件读取路径，当无需将墙体转为伺服墙体时，可采用默认值 None。 |
| | FileType [str] [option] | 设定墙体文件类型，可选为“NPZ”或“TXT”，默认值为“NPZ”。 |
| | Visualize [bool] [option] | 设定是否输出墙体可视化文件，当墙体类型为“Facet”或“Patch”时生效。默认值为 False。 |

(9) 颗粒本质属性

| | | |
|-----------------------------|-------------------------------------|---------------------|
| add_attribute(keywords ...) | | |
| Primary keywords: | | |
| ① materialID [int] | | |
| ② Attribute [dict] | Density [float] | 设置材料的密度。 |
| | ForceLocalDamping [float] [option] | 设置材料的局部阻尼系数，默认值为 0。 |
| | TorqueLocalDamping [float] [option] | 设置材料的局部阻尼系数，默认值为 0。 |

(10) 接触本构

| |
|------------------------------------|
| choose_contact_model(keywords ...) |
| Primary keywords: |

| | | |
|--|--|----------------------|
| ① <code>particle_particle_contact_model [str][option]</code> | | |
| | 设置颗粒-颗粒接触本构模型，可从以下四个模型中选择，若不存在颗粒与颗粒间的接触，可采用默认值 <code>None</code> 。 | |
| | Linear Model | 线性接触模型 |
| | Hertz Mindlin Model | 赫兹-明德林接触模型 |
| | Linear Rolling Model | 线性抗转动模型（Luding 2008） |
| | Jiang Rolling Model | 线性抗转动模型（Jiang 2013） |
| ② <code>particle_particle_contact_model [str][option]</code> | | |
| | 设置颗粒-墙体接触本构模型，可从以上四个模型中选择，若不存在颗粒与墙体间的接触，可采用默认值 <code>None</code> 。 | |

(11) 接触本质属性

| | | |
|----------------------------|--|----------|
| add_property(keywords ...) | | |
| Primary keywords: | | |
| ① materialID1 [int] | | |
| | 设置接触模型所属的材料 ID。 | |
| ② materialID2 [int] | | |
| | 设置接触模型所属的材料 ID。 | |
| ③ property [dict] | | |
| | 若在函数/choose_contact_model/中选择 Linear Model，需将下列关键词加入字典中，并赋值给 property 变量。 | |
| | NormalStiffness [float] | 法向接触刚度 |
| | TangentialStiffness [float] | 切向接触刚度 |
| | Friction [float] | 摩擦系数 |
| | NormalViscousDamping [float] | 法向粘滞阻尼系数 |
| | TangentialViscousDamping [float] | 切向粘滞阻尼系数 |
| | 若在函数/choose_contact_model/中选择 Hertz Mindlin Model，需将下列关键词加入字典中，并赋值给 property 变量。 | |
| | ShearModulus [float] | 剪切模量 |
| | Possion [float] | 泊松比 |

| | | |
|---|--|-------------|
| | Friction [float] | 摩擦系数 |
| | Restitution [float] | 恢复系数 |
| 若在函数/choose_contact_model/中选择 Linear Rolling Model, 需将下列关键词加入字典中, 并赋值给 property 变量。 | | |
| | NormalStiffness [float] | 法向接触刚度 |
| | TangentialStiffness [float] | 切向接触刚度 |
| | RollingStiffness [float] | 旋转接触刚度 |
| | TwistingStiffness [float] | 扭转接触刚度 |
| | Friction [float] | 摩擦系数 |
| | RollingFriction [float] | 旋转摩擦系数 |
| | TwistingFriction [float] | 扭转摩擦系数 |
| | NormalViscousDamping [float] | 法向粘滞阻尼系数 |
| | TangentialViscousDamping [float] | 切向粘滞阻尼系数 |
| | RollingViscousDamping [float] | 旋转粘滞阻尼系数 |
| | TwistingViscousDamping [float] | 扭转粘滞阻尼系数 |
| 若在函数/choose_contact_model/中选择 Jiang Rolling Model, 需将下列关键词加入字典中, 并赋值给 property 变量。 | | |
| | YoungModulus [float] | 杨氏模量 |
| | StiffnessRatio [float] | 切向刚度与法向刚度之比 |
| | Friction [float] | 摩擦系数 |
| | ShapeFactor [float] | 形状系数 |
| | CrushFactor [float] | 破碎系数 |
| | NormalViscousDamping [float] | 法向粘滞阻尼 |
| | TangentialViscousDamping [float] | 切向粘滞阻尼 |
| ④ dType [str] [option] | | |
| | 接触材料参数赋值对象, 可选为“particle-particle”、“particle-wall”或“all”, 默认值为“all”。 | |

(12) 存储数据类型

| |
|--------------------------------|
| select_save_data(keywords ...) |
|--------------------------------|

| | |
|---|---------------------------|
| Primary keywords: | |
| ① particle [bool] [option] | 是否存储球形颗粒信息，默认值为 True。 |
| ② clump [bool] [option] | 是否存储多球颗粒信息，默认值为 True。 |
| ③ wall [bool] [option] | 是否存储墙体信息，默认值为 False。 |
| ④ particle_particle_contact [bool] [option] | 是否存储颗粒-颗粒接触信息，默认值为 False。 |
| ⑤ particle_wall_contact [bool] [option] | 是否存储颗粒-墙体接触信息，为 False。 |

(13) 模型计算

| | |
|--|--|
| run(keywords ...) | |
| Primary keywords: | |
| ① visualize [bool] [option] | 在模拟过程中是否开启 GUI，实时渲染模拟结果，默认值为 False。 |
| ② callback [taichi.kernel] [option] | 进行真三轴试验时，需要在伺服过程中实时更新有效接触面积，若无需进行伺服时可采用默认值 None。 |
| ③ function [function / taichi.kernel] [option] | 用户自定义函数，可在每次循环时均调用该函数，无需此功能时可采用默认值 None。 |

(14) 颗粒属性更新

| | |
|--|------------------------------|
| update_particle_properties(keywords ...) | |
| Primary keywords: | |
| ① particle_type [str] | 设置颗粒的类型，可选为“Sphere”或“Clump”。 |

| | |
|---------------------------------------|---|
| ② property_name [str] | 设置需要修改属性的名称，可选为“bodyID”、“groupID”、“radius”、“position”、“velocity”、“angular_velocity”、“orientation”、“fix_velocity”或者“fix_angular_velocity”。 |
| ③ value [int / float / list] | 设定需要修改的参数，根据的属性名称，输入的类型也不同。 |
| ④ override [bool] [option] | 是否需要覆盖原先已有的属性，默认值为 False。 |
| ⑤ bodyID [int] [option] | 利用关键词“bodyID”过滤所需修改的颗粒，默认值为 None。关键词“bodyID”、“region_name”或“function”必须有一个为非 None。 |
| ⑥ region_name [str] [option] | 利用在函数/add_region/中预设的区域所需修改的颗粒，默认值为 None。关键词“bodyID”、“region_name”或“function”必须有一个为非 None。 |
| ⑦ function [taichi.function] [option] | 利用用户自定义的函数过滤所需修改的颗粒，默认值为 None。关键词“bodyID”、“region_name”或“function”必须有一个为非 None。 |

（15）墙体属性更新
尚未开发。

（16）接触属性更新

| | |
|---|---------------------|
| update_contact_properties(keywords ...) | |
| Primary keywords: | |
| ① materialID1 [int] | 设置所需修改接触属性对应的材料 ID。 |

| | |
|----------------------------|---|
| ② materialID2 [int] | 设置所需修改接触属性对应的材料 ID。 |
| ③ property_name [str] | 设置需要修改属性的名称，根据求解器中预设的接触模型，选择该接触模型对应的参数。 |
| ④ value [float] | 设定需要修改的参数。 |
| ⑤ override [bool] [option] | 是否需要覆盖原先已有的属性，默认值为 False。 |

（17）求解器参数更新

| | |
|------------------------------------|------------------------|
| modify_parameters(keywords ...) | |
| Primary keywords: | |
| ① Timestep [float] [option] | 修改模拟的时间步长，默认为初始设定值。 |
| ② CFL [float] [option] | 修改模拟的 CFL 系数，默认为初始设定值。 |
| ③ AdaptiveTimestep [bool] [option] | 修改是否实行自适应时间步，默认为初始设定值。 |
| ④ SaveInterval [float] [option] | 修改数据存储时间间隔，默认为初始设定值。 |
| ⑤ SavePath [str] [option] | 修改存储数据路径，默认为初始设定值。 |

（18）伺服开关

| | |
|----------------------------|-------------------------------------|
| servo_switch(keywords ...) | |
| Primary keywords: | |
| ① status [str] [option] | 修改伺服机制是否启动，可选为“On”或者“Off”，默认值为“On”。 |

（19）重新启动

| | |
|----------------------------|--|
| read_restart(keywords ...) | |
| Primary keywords: | |

| | |
|-------------------------------|------------------------------|
| ① file_number [int] | 需要重新读取的状态文件 ID。 |
| ② file_path [str] | 需要重读取的文件路径。 |
| ③ particle [bool] [option] | 是否需要读取颗粒的状态信息，默认为 True。 |
| ④ sphere [bool] [option] | 是否需要读取球体颗粒的状态信息，默认为 True。 |
| ⑤ clump [bool] [option] | 是否需要读取多球颗粒的状态信息，默认为 False。 |
| ⑥ wall [bool] [option] | 是否需要读取墙体的状态信息，默认为 True。 |
| ⑦ servo [bool] [option] | 是否需要读取伺服墙体的状态信息，默认为 False。 |
| ⑧ ppcontact [bool] [option] | 是否需要读取颗粒-颗粒接触的状态信息，默认为 True。 |
| ⑨ pwcontact [bool] [option] | 是否需要读取颗粒-墙体接触的状态信息，默认为 True。 |
| ⑩ is_continue [bool] [option] | 是否输出文件 ID 是否延续，默认为 True。 |

(20) 后处理

| | |
|---------------------------------|--------------------------------------|
| postprocess(keywords ...) | |
| Primary keywords: | |
| ① start_file [int] [option] | 设置读取起始文件 ID，默认值为 0。 |
| ② end_file [int] [option] | 设置读取结束文件 ID，默认值为本次模拟结束时最后存储的文件 ID。 |
| ③ read_path [str] [option] | 设置读取文件的路径，默认值为“./OutputData”。 |
| ④ write_path [str] [option] | 设置输出可视化文件的路径，默认值为“./OutputData/vtk”。 |
| ⑤ write_bodyID [bool] [option] | 是否在可视化文件中输出颗粒 ID，默认值为 True。 |
| ⑥ write_groupID [bool] [option] | 是否在可视化文件中输出颗粒所属组的 ID，默认值为 True。 |

| | |
|--|----------------------------------|
| ⑦ write_radii [bool] [option] | 是否在可视化文件中输出颗粒的半径，默认值为 True。 |
| ⑧ write_displacement [bool] [option] | 是否在可视化文件中输出颗粒的位移及其分量，默认值为 True。 |
| ⑨ write_velocity [bool] [option] | 是否在可视化文件中输出颗粒的速度及其分量，默认值为 True。 |
| ⑩ write_angular_velocity [bool] [option] | 是否在可视化文件中输出颗粒的角速度及其分量，默认值为 True。 |
| ⑪ write_force_chain [bool] [option] | 是否在可视化文件中输出力链，默认值为 False。 |
| ⑫ write_wall [bool] [option] | 是否在可视化文件中输出墙体，默认值为 False。 |

3. 案例及运行结果

本节将展示两个验证算例，以证明的多功能性与高效性，所有验证算例均在 Ubuntu 20.04 系统上进行,采用 AMD AM4 5800X CPU(8 核@3.8GHz)和 GeForce RTX 3070 GPU。

3.1 三轴排水试验

本算例中考虑了三种不同初始密实的颗粒集合体来验证 *GeoTaichi* 离散元模块。在立方体容器内,生成 28482 个球形颗粒,颗粒的粒径范围为 0.25mm~0.55mm, 粒径服从均匀分布。材料参数取值：接触刚度由经验取值为 $k_n = k_s = r \times 100MPa$ ，其中 r 为颗粒平均半径。在固结阶段颗粒间摩擦系数分别为 0, 0.2 及 0.5，分别生成初始孔隙比为 0.550、0.707 及 0.747 的试样。在剪切阶段，上下板按照既定速度运动，使得试样受剪切，且满足准静态条件。在此阶段摩擦系数取为 0.5。图 3 展示了初始孔隙比为 0.550 的试样在剪切过程中颗粒位置及力链的演化过程。

试样的应力张量可以由颗粒间接触力定义：

$$\boldsymbol{\sigma} = \frac{1}{V} \sum_{c \in V} \mathbf{f}^c \mathbf{l}^c \quad (1)$$

其中 V 为试样总体积, \mathbf{f}^c 为接触 c 处的接触力, \mathbf{l}^c 为接触支向量。平均应力 p 和偏应力 q 可以由下式表达:

$$p = \frac{1}{3} \text{tr}(\boldsymbol{\sigma}) \quad q = \sqrt{\frac{3}{2} \boldsymbol{\sigma}':\boldsymbol{\sigma}'} \quad (2)$$

其中 $\boldsymbol{\sigma}'$ 为应力偏张量。试样的体积应变由 $\epsilon_v = \ln(V/V_0)$ 计算, 其中 V_0 和 V 分别为试样的初始和最终体积, 且体积应变为正值代表试样发生剪胀。

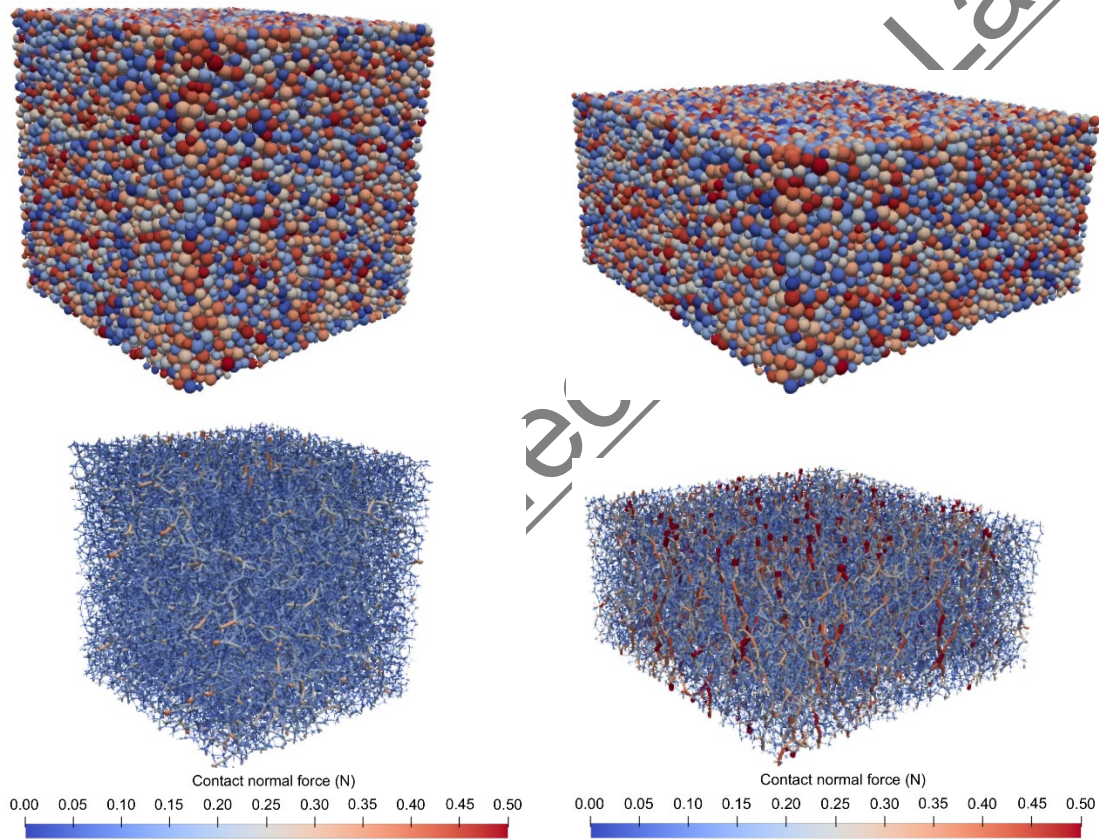


图 3 三轴排水颗粒空间分布及力链演化

图 4 展现了三种试样在三轴排水条件下的宏观响应。所有试样在最终均达到了统一的临界状态, 即达到稳定的应力比 q/p 和稳定的体积应变。对于密实试样而言, 在剪切过程中表现出了应力软化及剪胀现象, 而松砂则一直表现出应力硬化及剪缩的现象。因此模拟结果与土力学中著名的砂土实验观测结果定性一致, 证明了 *GeoTaichi* 数值计算的有效性和鲁棒性。

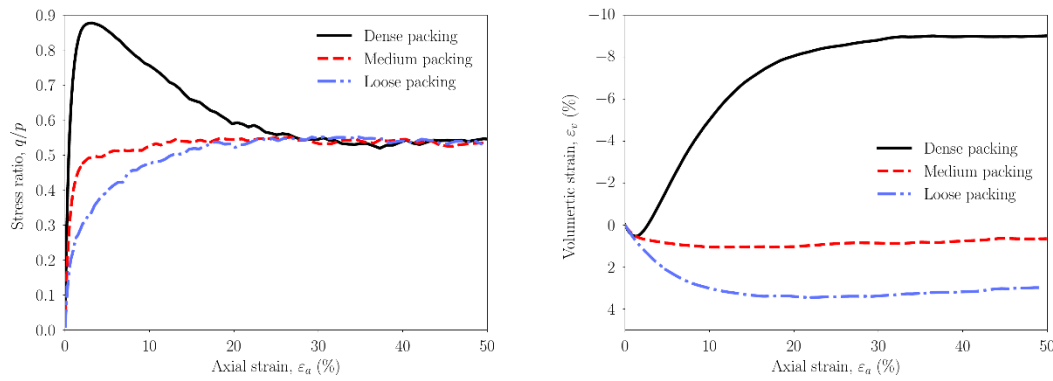


图 4 三轴排水宏观应力比及体积应变演化

3.2 自然休止角

在物质点—离散元耦合方法中，接触检测是需要被重点关注的计算模块。因为该模块在离散元颗粒接触及物质点—离散元接触力中起到重要作用，同时也是计算耗时最大的模块。因此本节采用大规模颗粒柱倒塌算例来验证 *GeoTaichi* 中的计算效率，并将其与现有基于 GPU 的高效离散元计算软件 *Musen* 进行对比。如图 5 所示，首先在 $15\text{ m} \times 15\text{ m} \times 15\text{ m}$ 的立方体中生成指定的颗粒数量的颗粒柱，使其在重力作用下自由下落，最终在平面上堆积（图 6）。颗粒间接触采用赫兹模型，剪切模量为 4.3×10^6 ，泊松比为 0.3，摩擦系数为 0.5，恢复系数为 0.6。

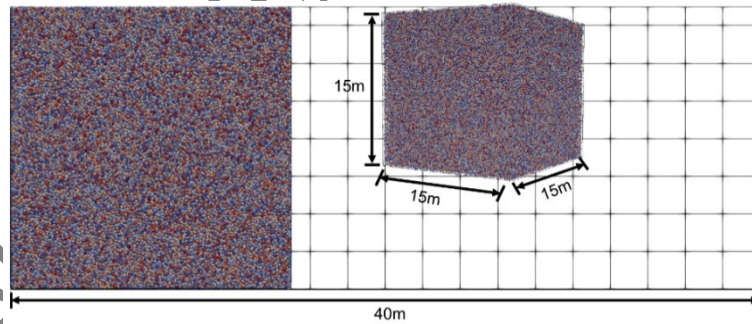


图 5 颗粒柱坍塌初始状态

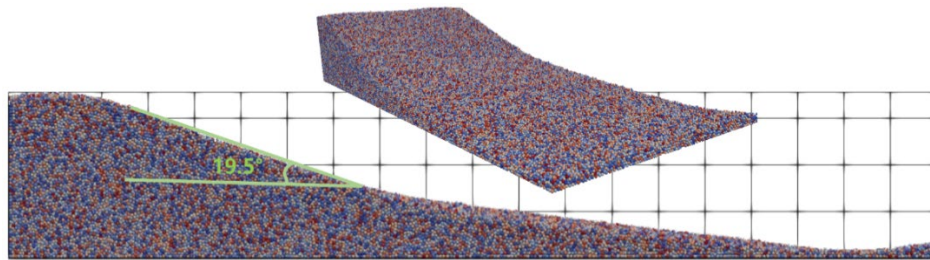


图 6 颗粒柱坍塌最终形态

首先考虑了不同松弛系数($\lambda = 0, 0.1, 0.2, 0.3, 0.4$)下, *GeoTaichi* 的计算效率。由图 7 可以看出, 在开始计算阶段, 由于配位数不断增多导致离散元的计算效率迅速下降。当松弛系数为 0 时, 即 Verlet 列表在每个时间步均更新, 此时计算效率最低。随着松弛系数的增大, 离散元计算效率逐渐提高, 当松弛系数超过 0.2 时, 其计算效率接近收敛不再提高。因此在下列算例中, 松弛系数均取 0.2。

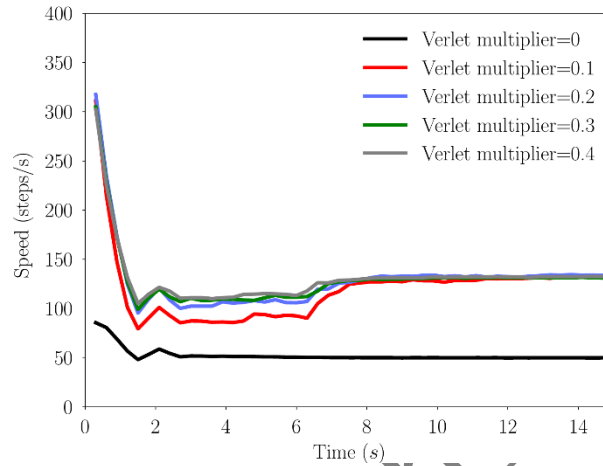


图 7 不同松弛系数下离散元计算效率变化

此外也考虑了不同规模(颗粒数为 0.1M, 0.3M, 0.6M, 1M 和 1.6M)下离散元的计算效率, 图 8 可以看出随着颗粒的增加, 计算效率在不断降低。并且也考虑了不同 GPU 型号对计算效率的影响。RTX4080 的核心数和内存最大, 其次为 RTX3070, 最小为 RTX2070。可以看出随着 GPU 核心和内存的增加, 离散元计算效率不断提高。

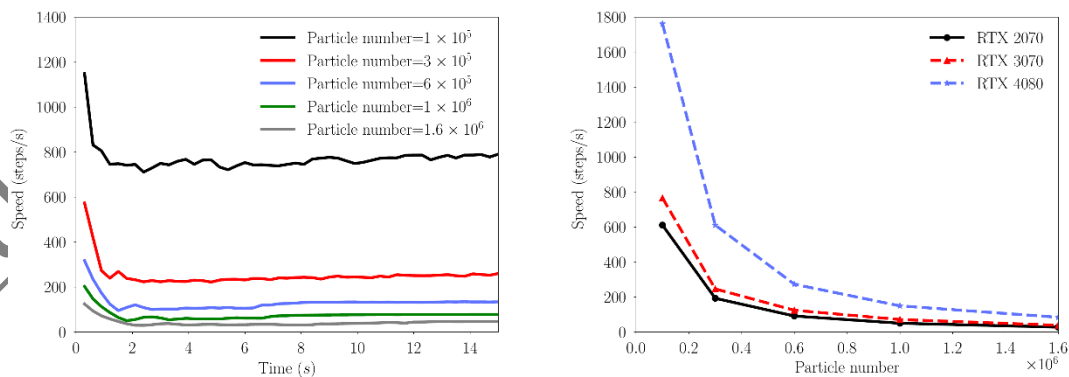


图 8 不同颗粒规模及 GPU 型号下离散元计算效率变化

图 9 将 *GeoTaichi* 的计算效率与基于 CPU-GPU 异构并行的离散元开源软件

Musen 进行了对比。可以看出 *GeoTaichi* 的计算效率远远高于 Musen，特别是当颗粒规模较小的时候。这可能是由于在 CPU-GPU 异构并行时，CPU 与 GPU 之间的数据传输耗时较大，使得 GPU 计算性能未被完全利用。此外可以看出相同颗粒规模下，*GeoTaichi* 所占内存远远小于 Musen，这更有利于解决 GPU 显存过小的瓶颈，使得大规模计算成为可能。

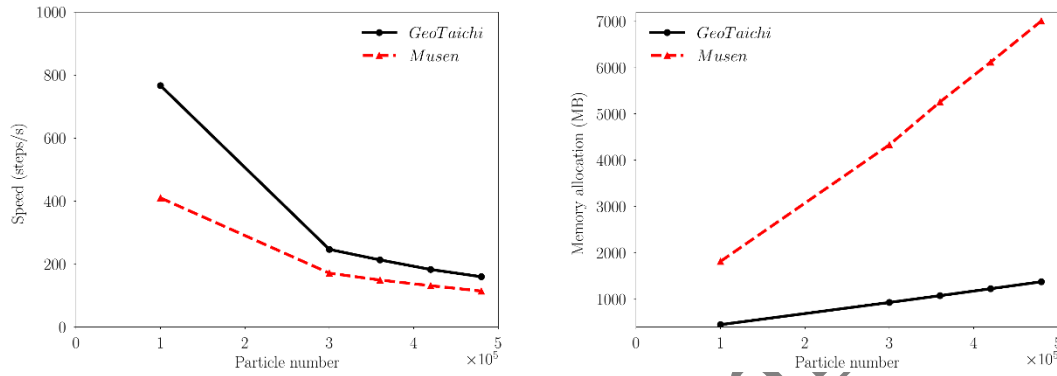


图 9 *GeoTaichi* 与 Musen 计算性能及内存占用对比