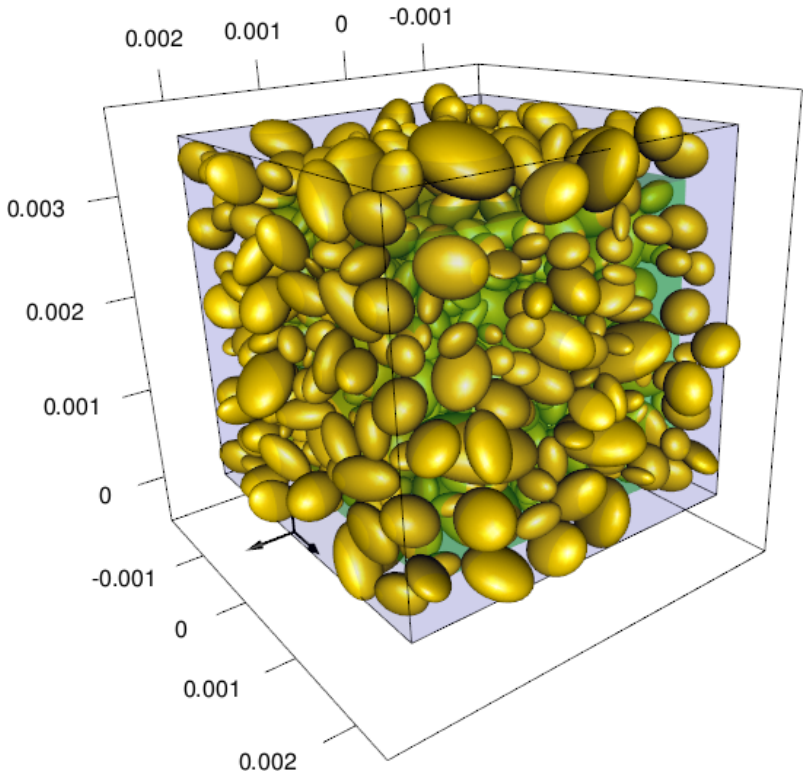
GranularSim User Guide

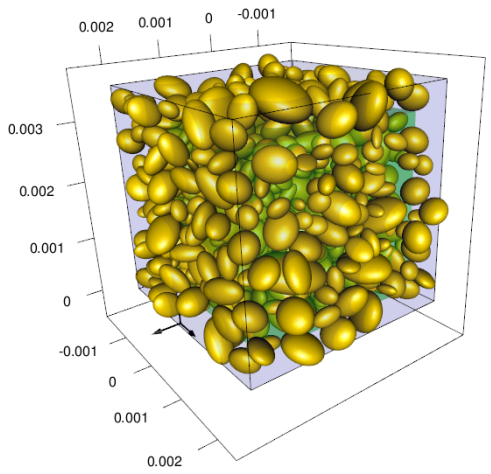
Version Version 18 .2

February 3, 2018

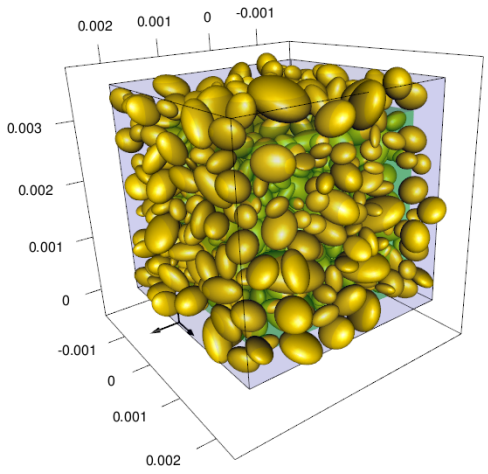


Biswajit Banerjee

The contents of this manual can and will change significantly over time. Please make sure that all the information is up to date.



Contents



1 — Overview of GranularSim

GranularSim is a rewritten and updated version of the parallel discrete element code ParaEllip3D developed at the University of Colorado at Boulder, with the intent of having a more general code that can be improved and appended to more easily. The primary use case is the simulation of soils and other granular materials.

The main components of GranularSim include a discrete element (DEM) code that can handle ellipsoidal particles, a peridynamics (PD) code that couples to DEM particles, and a smoothed particle hydrodynamics (SPH) code that also couples to DEM particles. In addition, the code is able to handle special boundary conditions that are needed for hierarchical upscaling.

GranularSim provides outputs in VTK format and results can be visualized either with ParaView (https://www.paraview.org/) or with VisIt (https://wci.llnl.gov/simulation/computer-codes/visit/).

1.1 Downloading and compiling the Code

For instructions on downloading and compiling GranularSim see the Installation Guide which can be downloaded from https://github.com/bbanerjee/ParSim/tree/master/GranularSim/Manuals/InstallationGuide.

1.2 Running the code

> We will assume that you have compiled GranularSim inside a directory called opt.

In the opt directory you will find the executable granularsim. This is the entry point for the simulation code.

A typical simulation can be run either on one processor or on multiple processors. For a single processor run, use

```
cd opt
./granularsim <input_file.xml>
```

For a multiprocessor run, use

```
cd opt
mpirun -np <num_processors> ./granularsim <input_file.xml>
```

On Cray machines, you may have to use a PBS script to runs jobs. For jobs on distributed computing machines, a common replacement for mpirun is aprun . For interactive jobs on such machines you will need to use the alternative command

```
cd opt
aprun -n <num_processors> ./granularsim <input_file.xml>
```

1.3 Input files

GranularSim does not have a user interface that you can use to generate input files. Instead, you will have to create these files yourself. The input files are is XML format. A snippet from such a file is given below.

```
<?xml version='1.0' encoding='ISO-8859-1' ?>
<Ellip3D_input>

  <Meta>
     <title> DEM-PD simulation sample input file </title>
  </Meta>

  <!-- #simulation type -->
  <SimulationType>  001 </SimulationType>

  <!-- Parallel setup -->
  <Parallel>
    <!--
    #grids/processes in x, y, z directions for MPI parallelization, which
    #is recommended for more than thousands of particles. By default these
    #values are 1 for serial computing. They must satisfy equation:
    #mpiProcX * mpiProcY * mpiProcZ = np, where np is the number of processes
    #specified in mpirun command line.
    -->
    <mpiProc> [2, 1, 2] </mpiProc>

    <!--
    #threads per process for OpenMP parallelization, which is recommended
    #for more than thousands of particles. By default this value is 1 for
    #single-thread computing. If it > 1, multiple-threads computing is
    #invoked.
    -->
    <ompThreads>  1 </ompThreads>
  </Parallel>
  .......
</Ellip3D_input>
```

Further details can be found in the chapter on input files Add link to chapter.

1.4 Output files

GranularSim simulations produce a number of output files for each timestep that is saved. Each simulation produces a new directory that contains a set of files that are numbered sequentially.

> At present, all important variables are saved and selective saves are not allowed.

The files that are generated are:

1. bdry_contact_nnnnn.csv and bdry_contact_nnnnn.xml : Information about boundary contacts.
2. boundary_nnnnn : Information about the positions of the domain boundaries.
3. contact_nnnnn.csv and contact_nnnnn.xml : Information about inter-particle contacts.
4. domain_nnnnn.vtu : VTK format files containing the positions of domain boundaries.
5. oriented_domain_nnnnn.vtu : VTK format files containing the positions of domain boundaries for domains (including those that are not axis-aligned).

6. output_particles_nnn.dat : CSV format files containing the particle information.
7. particle_nnnnn.vtu : VTK format files containing the particle information.
8. patchGrid_nnnnn.vtu : VTK format files containing the domain decomposition information (patches).

Further details can be found in the chapter on output files Add link to chapter.

1.5 Visualization

The output generated by GranularSim can be visualized in three ways:

1. Using a R script.
2. Using ParaView to read and visualize the VTK output files.
3. Using VisIt to read and visualize the VTK output files.

Details can be found in the chapter on visualization Add link to chapter.