The Snowflake-Toolkit Updates

Torbjörn Rathsman

November 9, 2016

The Snowflake-Toolkit is a software toolkit intended to generate shape data for simulating microwave scattering from snow and ice particles. This paper briefly describes changes since the October 2016

Contents

1 Graupel generator and sphere aggregates

The toolkit now contains a graupel generator `graupel_generate2`, described in [1]. This required an additional geometry representation, since it was too inefficient to use polygon meshes when building large aggregates of spheres. The new representation takes advantage of the fact that the sub-volumes are all spheres. This makes the data structures simpler, and reduces the amount of work needed for hit-testing, overlap detection, and ray-casting. Also, overlapping volumes are trivial to compute. This means that all computed volumes should be exact. A drawback is that $D_{\max}$ requires more effort to compute exactly. Thus, it is approximated by comparing the 6 axis-aligned points on each sphere.

2 Rewrite of rasterizer

The addition of an additional aggregate representation motivated a rewrite of the rasterizer. The previous callback-based approach has been replaced by a direct approach. The floodfill method of the aggregate object is given a `Grid` object, and sets its bits directly. When the ADDA file is written, it loops through the filled grid, and writes the indices of all non-zero elements. The new approach should be an order of magnitude faster, due to the simplified call graph.

3 Visualisation of point clouds

Due to the large number of vertices resulting from generating icospheres from a sphere aggregate, visualisation becomes demanding. To overcome this problem a very bare-metal tool `addaview`, for visualising a point cloud has been written. The tool uses nothing more than OpenGL 3.3+ to render a set of points. It features some primitive mouse and keyboard navigation for switching between perspective and orthographic view, rotate the cloud, and render slices.

4 Support for mixture of prototypes in aggregate generator

The aggregate generator has been extended to support custom mixture of prototypes, and probability distribution. A mixture is a set of deformation rules, each with an associated crystal prototype and probability. Mixtures are stored in JSON format. The outer object is an array where each object has the properties `prototype`, `probability`, and `deformations`. The latter property is an array of deformation rules, which has the same properties as they had before. Each rule is encoded as an array, with the ordered elements *parameter*, *distribution*, and *distribution parameters*.

The mixture file refers to other files, prototypes, and optionally custom probability distributions. The paths of these files are assumed to be relative to the loaction of the mixture file. This design choice is motivated by the assumption that mixture files, and the files they refer to, together forms a database, and that database should be usable regardless of current working directory.

5 MATLAB and Octave frontend

To make it easier to integrate the toolkit into the workflow, a set of wrapper functions for MATLAB and Octave has been written. These functions makes it easier to start a simulation, and when it is done, analyse its output. Other features include mixture file generation, and a function to start `addaview`. It is also possible to import the mesh contents of an `ice` file.

6 Shadow mask renderer

The toolkit now features two programs for rendering shadow masks of meshes, and ADDA files. These programs are implemented using OpenGL 3.3 and a simple orthographic projection. The programs both writes the result into a PNG file, with the resolution set to the number of pixels per length unit, multiplied by 1024. There exists MATALB and Octave frontents as well.

7 External libraries required by the snowflake toolkit

This is a list of all external libraries used by the toolkit, as reported by `maike`.

- `GL`
- `hdf5`
- `hdf5_cpp`
- `png`
- `pthread`
- `hdf5_hl_cpp`
- `glfw`
- `GLEW`
-
- jansson

References

[1] `https://github.com/milasudril/snowflake-toolkit/blob/master/changelog/graupelgen/graupelgen.pdf`