Compute the new position according to the wind speed at the location of the particle.

Compute the initial particles according to the input mesh.

Implement move the particles around the building. Mark all the tetrahedron adjacent to the building. If the particle try to enter the building, project its velocity to the out face to get the new velocity.

Refractor the data structure using 1D array instead of 2D array. Convert the source data file to a binary file.

10\_9

Init the particles in a sphere and box.

Move the sphere source easily.

Improved the user interface for navigation.

Fix the key event handler.

10\_13

Extend it to 3D.

Generate the particles batch by batch.

The number of batches can be selected by the user.

The number of paths displayed can be selected by the user.

3D button can be shadowed when pressed.

Update the tet intersected with the sphere only when the mouse button is release. It is also updated when the mouse is dragging and the particles are generated from the sphere.

Add the button to switch the mode to move the view point, the light or the source.

Add the button to hide the particles.

Add a 3D target to let the user to specify the focus plane interactively.

10\_22

Specify an active area using a box. The user can change the size and position easily.

Display the batch info in the status bar.

Updates the age of the particles. If the age reach the lifetime, reset the particle.

Only deal with the initCount particles when sort and display them.

Next

Add a initial velocity. When computing the new velocity, add the wind in the new position and the old velocity, scale the sum vector, change its length to the that of the wind in the new position.

Improve the FindinSphere function. Implement it on GPU.

When Init the particles in the sphere, init a particle randomly in the sphere, and find out which tet contains it.