**Stable Fluids Algorithm**

**Setup**

* Number of Dimensions (2 or 3)
* Origin of grid
* Length of grid (of each side) [L]
* Number of cells [N]
* Voxel size [L/N]
* Two grids for each component of the velocity at time steps ti & ti+1 [U0, U1]
* Two grids for scalar field (density) at time steps ti & ti+1 [S0, S1]
* Time step [dt]
* Viscosity [visc]
* Diffusion Constant [ks]
* Dissipation rate [as]
* Boundary Conditions (simplest is 0)

**Simulator**

While ( simulating ) {

/ \* handle display and user interaction \*/

/ \* get forces F and sources Ssource from UI \*/

Swap (U1, U0);

Swap(S1, S0);

Vstep ( U1, U0, visc, F, dt);

Sstep ( S1, S0, kS, aS, U1, Ssource, dt);

}

Vstep ( U1, U0, visc, F, dt ) {

for(i=0; i<NDIM; i++)

 addForce ( U0[i], F[i], dt );

for(i=0; i<NDIM; i++)

 Transport ( U1[i], U0[i], U0, dt);

for(i=0; i<NDIM; i++)

Diffuse ( U0[i], U1[i], visc, dt);

Project ( U1, U0, dt );

}

Sstep ( S1, S0, k, a, U, source, dt ) {

addForce ( S0, source, dt );

Transport ( S1, S0, U, dt );

Diffuse ( S0, S1, k, dt );

Dissipate ( S1, S0, a, dt );

}