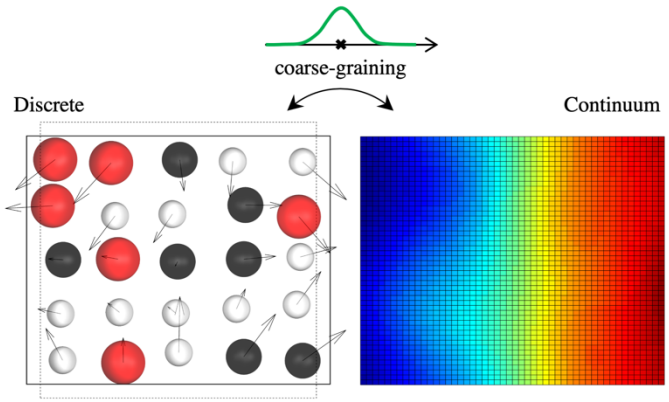


Download tutorial

Introduction

The technique discussed in this tutorial is a weighted Gaussian convolution or the so-called coarse-graining method. Its aim is to transform discrete quantities in space to continuum variables. We can use this method to establish the correlation between discrete simulations like molecular dynamics or discrete elements, and a continuum description (e.g. finite element method). Present method should not be confused with coarse-grained molecular simulations, where stiff atomic configurations (e.g. pyran rings) are replaced by a single larger particle.

The algorithm presented here is a generalized version of the code used in Ref. [1], [2] and [3].

The coarse graining function

The averaging function was choosen as the following form :

$$\phi(r) = \frac{1}{w^n \pi^{n/2}} e^{-(r/w)^2}, \qquad (1)$$

where *w* is the coarse-graining width and *n* is the number of dimensions.

The function of normalized in order to obtain a unit integral in an infinite domain:

$$\int_{\Omega} \phi(r) d\Omega = 1, \qquad (2)$$

with the distance between grid point and particle:

$$r = \left\| \mathbf{r}_i - \mathbf{r} \right\| = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}, \qquad (3)$$

and *x*, *y* and *z* are the coordinates of the grid point, while x_i, y_i and z_i are the coordinates of particle i.

Basic concept

To demonstrate the principal idea of the coarse-graining, consider the following 1D example:

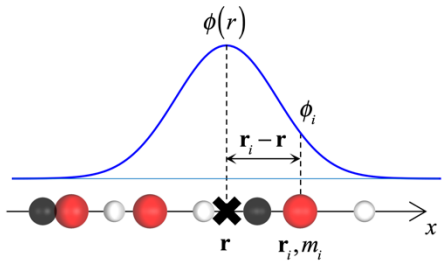


Figure 1 – 1D concept of coarse-graining.

To obtain the mass belonging to the grid point (highlighted using a black cross) the following equation is calculated:

$$\rho^{CG}(\mathbf{r}) = \sum_i m_i \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right) = \sum_i m_i \phi_i . \tag{4}$$

Each particle is taken into account with its proper mass and a weight associated with the particle as a function of its distance to the grid point.

This technique can also be used for stress, energy or other discrete quantities defined on particles.

Coarse-graining displacement

To calculate the displacements on grid points, the particle displacement is normalized by its mass/volume. The weighted average is obtained by the following equation:

$$\mathbf{u}^{CG}(\mathbf{r}) = \frac{\sum_i \mathbf{u}_i m_i \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)}{\sum_i m_i \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)}, \tag{5}$$

where $\mathbf{u}_i$ is the displacement vector of atom *i*.

Spatial derivatives (e.g. strains)

The spatial derivatives of the displacements can be calculated analytically. The calculation on the x directional deformation is used to demonstrate the technique:

$$\begin{aligned}
\varepsilon_x(\mathbf{r}) &= \frac{\partial u_x}{\partial x} = \frac{\partial}{\partial x}\frac{\sum_i u_{x,i} m_i \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)}{\sum_i m_i \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)} = \frac{\partial}{\partial x}\frac{\sum_i u_{x,i} m_i \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)}{\rho(\mathbf{r})} \\
&= \frac{\rho(\mathbf{r})\frac{\partial}{\partial x}\left[\sum_i u_{x,i} m_i \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)\right] - \left[\sum_i u_{x,i} m_i \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)\right]\cdot\frac{\partial \rho(\mathbf{r})}{\partial x}}{\rho(\mathbf{r})^2} \\
&= \frac{\sum_i u_{x,i} m_i \frac{\partial \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)}{\partial x}}{\rho(\mathbf{r})} - \frac{\left[\sum_i u_{x,i} m_i \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)\right]\cdot\frac{\partial \rho(\mathbf{r})}{\partial x}}{\rho(\mathbf{r})\cdot\rho(\mathbf{r})} \\
&= \frac{\sum_i u_{x,i} m_i \frac{\partial \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)}{\partial x}}{\rho(\mathbf{r})} - u_x^{CG}(\mathbf{r})\cdot\frac{\sum_i m_i \frac{\partial \phi\left(\left\|\mathbf{r}_i - \mathbf{r}\right\|\right)}{\partial x}}{\rho(\mathbf{r})}.
\end{aligned} \tag{6}$$

The spatial derivatives of the coarse graining function, can be obtained as:

$$\begin{aligned}\frac{\partial\phi(r)_i}{\partial x} &= \frac{1}{w^3\pi^{3/2}}\frac{\partial}{\partial x}\left[e^{-(r^2/w^2)}\right] \\ &= \frac{1}{w^3\pi^{3/2}}e^{-(r^2/w^2)}\cdot\frac{\partial}{\partial x}\left(-\frac{r^2}{w^2}\right) \\ &= \frac{1}{w^3\pi^{3/2}}e^{-(r^2/w^2)}\cdot\left(-\frac{1}{w^2}\right)\cdot\frac{\partial}{\partial x}(x-x_i)^2 \\ &= \frac{1}{w^3\pi^{3/2}}e^{-(r^2/w^2)}\cdot\left(-\frac{1}{w^2}\right)\cdot 2(x-x_i) \\ &= -\frac{2(x-x_i)}{w^5\pi^{3/2}}e^{-(r^2/w^2)}.\end{aligned} \qquad (7)$$

Example

To execute the coarse-graining script, a MATLAB version 2016 or later is required.

The input files for a simple 0.5% shear in the xy plane and the coarse-graining script can be downloaded Here.

To start the coarse-graining the following command should be executed:

```
[NODES,CG]=CGtoGRID(ini,def,cgwidth,nx,ny,nz,zviz,cutoff)
```

Where:

`ini` – is the name of the MATLAB file containing the reference configuration.
`def` – is the name of the MATLAB file containing the deformed configuration.
`cgwidth` – is the coarse-graining width (*w* in eq. 1).
`nx,ny,nz` – are the number of grid segments (grid point number = ni+1).
`zviz` – slice number to evaluate in the z direction (if zviz = 0 we calculate all nz).
`cutoff` – coarse-graining cut-off distance (usually 3*w*).

The files containing the reference and deformed configurations should contain 3 variables:

`Coord` – a Nx5 size array, where N is the number of particles:

Serial number of particles	X coordinate	Y coord.	Z coord.	Weight
1	2.4	16.7	-22.3	28
2	-38.8	-18.7	-1.5	28
…				

`Box` – a 3x3 size array containing the bons of the simulation box:

x lower bond	x upper bond	xy tilt
y lower bond	y upper bond	xz tilt
z lower bond	z upper bond	yz tilt

`Data` – is a NxM array containing M discrete information on each atom (e.g., stress, energy, etc.)

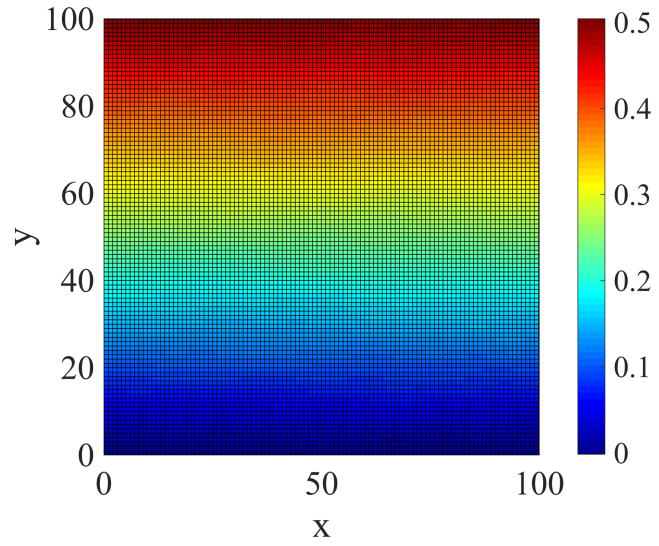
For example to execute the script in this case you can use:

```
[NODES,CG]=CGtoGRID('ini.mat','def.mat',5,100,100,100,50,15)
```

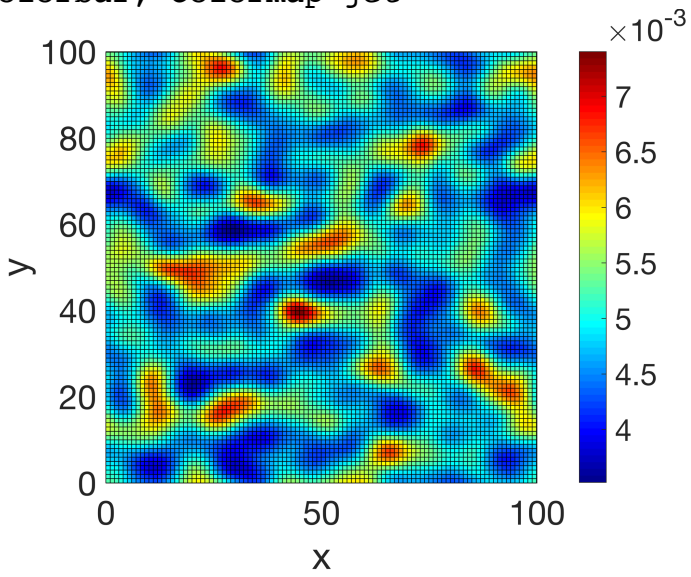
As a result the algorithm will provide nodal quantities in the `NODES` structure and coarse-grained values on a regular grid in the CG structure. In case `zviz=0`, a 3D array will be provided.

To visualize the displacement in the x direction, the following command can be used:

```
pcolor(CG.cgx,CG.cgy,CG.ux)
colorbar; colormap jet
```




```
pcolor(CG.cgx,CG.cgy,CG.exy)
colorbar; colormap jet
```



And the average strain corresponds well to the applied macroscopic displacement:

```
mean(CG.exy(:))

ans =

    0.0050
```

CAUTION
The sum between steps should be done progressively with strain steps smaller than 10%.

To cite this script, please refer to the following papers:

[1] G. Molnár, P. Ganster, J. Török, A. Tanguy, Sodium effect on static mechanical behavior of MD-modeled sodium silicate glasses, Journal of Non-Crystalline Solids, 440 pp. 12-25, 2016.

[2] A. Tanguy, Effect of composition and pressure on the shear strength of sodium silicate glasses: An atomic scale simulation study, Physical Review E, 95, 043001, 2017.

[3] G. Molnár, D. Rodney, P. Dumont, F. Martoïa, Y. Nishiyama, K. Mazeau, L. Orgéas, Cellulose crystals plastify by localized shear, Proceedings of the National Academy of Sciences of USA, 115 (28) 7260-7265, 2018.