Suspended Bodies and Cells

AC290r - Simone Melchionna

National Research Council Italy & Lexma Technology LLC

April 23, 2019

Project Updates

Extreme Computing: not for the faint of heart !

Projects Updates

Parallel partitioning is a challenge due to the system aspect ratio.

Group1:

- for Re=5, longitudinal size is 500, not easy to select partitioning and code crashes above 256 cores
- Re=10 and 15 should be ok for 1024 cores
- Issue with visualization, some glitch at domain frontiers: possibly corrected now

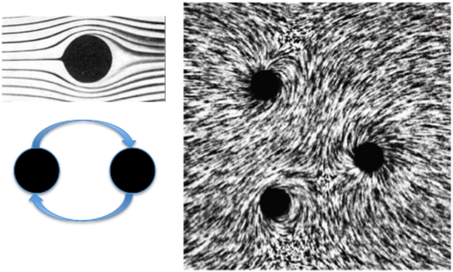
Group2:

- RBC excluded volume forces pose more challenges on domain splitting
- Use a periodic system
- Special absorbing BC for drug (see following slides)
- Run one simulation at Re=10 at zero hematocrit (~8M NODES)
- Run one simulation at Re=10 and for as long as computational resources allow (~8M NODES, ~1M RBC)

Recommendations: align with repository

LBM hosting RBC

The motion of a body in a fluid generates a disturbance in the surrouding and impacts other bodies.



Hydrodynamic interactions (HI) are solvent-mediated forces exerted by bodies in relative motion.
HI act over very long distances ($1/r$) and propagate with a characteristic time inversely proportional to viscosity.
HI are all-to-all in character and with the fluid solver we can compute them explicitly.

Coupling LBM fluid and particles

LBM can host N suspended point-like particles that exert local forces on the fluid:

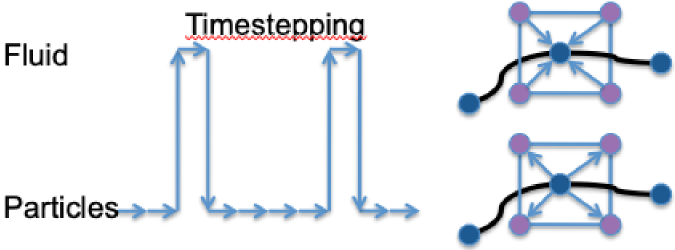
$$\begin{aligned} f_p(x+c_p, t+1) &= f_p(x,t) - \omega(f_p - f_p^{eq})(x,t) + \sum_{i=1}^{N} w_p \frac{\Lambda_i \cdot c_p}{c_s^2} \\ V_i(t+1) &= V_i(t) + \frac{1}{M} F_i - \Lambda_i \end{aligned}$$

A convenient momentum-preserving exchange kernel is:

$$\Lambda_i(x,t) = \gamma_T \left[V_i(t) - u(x,t) \right] \delta(x - R_i(t))$$

with γ_T: translational coupling coefficient.
The Dirac delta function cannot be represented numerically, but a number of approximate forms are available.

Information Exchange

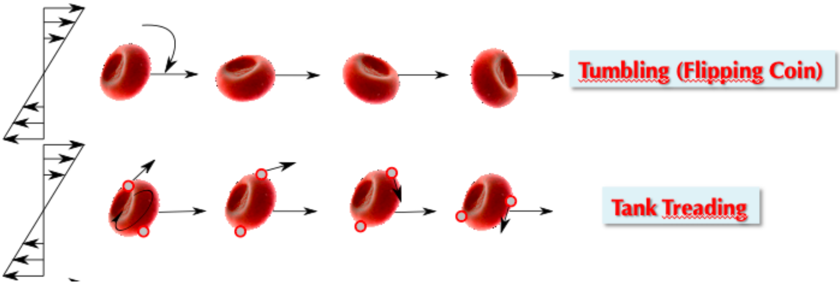


The simplest scheme couples particle and the nearest mesh point, a more sophisticated one uses a polynomial shape function.

The idea is to provide a continuous coupling as a particle moves in the continuum.
Available kernels must have special properties, such as invariance by translations and smoothness.

RBC-plasma coupling

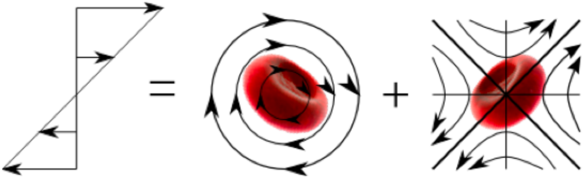
A RBC shows different types of motion under flow, in particular when this can be approximated as a linear shear flow.



A RBC will tuble and tank tread, as much as deforming if the shear forces are intense (eg in capillaries). For macrocirculation, tumbling and tank treading are sufficient as an accurate model.

Vorticity and elongational flow

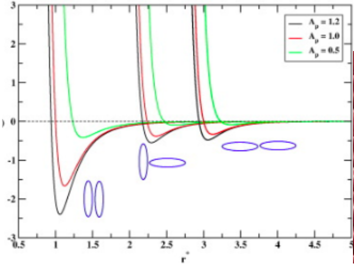
RBC rotation and deformation are caused by the underlying field. A linear shear field can be awlays decomposed in a rotational and elongational component.



Tumbling is a direct consequence of vorticity ($\omega = \nabla \times \mathbf{u}$) while stretching of elongational flow.
Orientational motion is controlled by the rotational coupling coefficient γ_R.

Excluded Volume

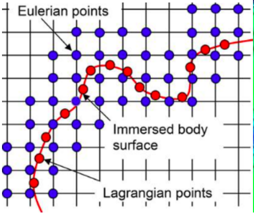
Constructing a contact force is not simple because of its singular behavior. A soft force is a much better object.
To emulate excluded volume, oarticles can be equipped with a repulsive force.

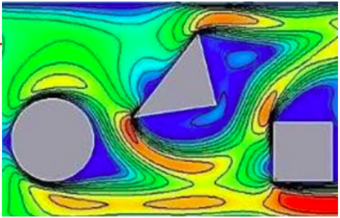


If the particle is non-spherical, analytical types of forces have been know for some time, e g the Gay-Berne type of forces.
in this way a small action radius is allowed.

More extended models

Immersed Boundary Method is the most used method to include arbitrarily deformable bodies.





Based on the idea of forces that expel fluid momentum, each lagrangian point acts on the surrounding fluid without imposing a boundary condition exactly. the method is more accurate with the higher density of lagrangian nodes.

Fluid-Structure Interaction

The name refers to interaction of some movable or deformable structure with an internal or surrounding fluid flow, e.g. to look into cells deforming in a capillary.

The Immersed Boundary Method is capable to include deformability: the lagrangian points organized as a mesh are subjected to structural forces.

By including stretching, bending and tensile mechanical response, we can model complex structural behaviour.

All forces have a direct impact on the surrouding fluid, in fact, the system is in equilibrium when all mechanical forces sum locally to zero.

The Newton–Raphson method or various fixed-point iteration are used to solve numerically the FSI conditions.

Simulation protocol for RBC

Starting a simulation with many RBCs is challenging:

- RBC have to be positioned at rather high packing fraction (up to 50% hematocrit), avoiding overlaps between RBCs and with wall
- A first part of simulation has to consider finding a good relaxed initial state (growth of excluded volume)
- Coupling with the fluid has to be gentle
- Having inlet + outlet poses further challenges, periodic is easier
- Parallelism is also more challenging since excluded volume forces require a minimal domain size

Nothing prevents the system from going wild.
The scripts prepare_run2_rbc.py in the repository illustrate a possible protocol.

RBC Protocol: Geometry construction

- Generate the confining geometry in absence of RBC
- For project, use a periodic geometry: see preproc1.py
- Insert RBC in the system: see preproc2.py
- Be sure that the partitioning fits the geometry

RBC Protocol: Simulation(1)

```
...
RESTART = ....
...
u.decorate()

# initial thin band of drug
nx,ny,nz = m.getBox()
nx = int(nx); ny = int(ny); nz = int(nz)
natms_tot = a.getNumberTotal()
myid = get_myproc()
profile2 = c.getArray(nx*ny*nz)
for k in range(1,nz+1):
    for j in range(1,ny+1):
        for i in range(1,nx+1):
            ifl = m.getLocator(i,j,k)
                # create BOLUS
                if i > nx/8 - 3 and i < nx/8.
+ 3:
                      profile2[ifl] = C1
                else:
                    profile2[ifl] = C0
c.setDensityProfile(profile2)
```

```
# freeze the fluid and drug until released
f.setFreeze(True)
c.setFreeze(True)

# set a robust friction coefficient for initial
equilibration
a.setCapForces(True, forcecap=1.e4,
torquecap=1.e4, velcap=0.4,
angvelcap=0.6)
a.setGamma(gammaT=0.1, gammaR=0.1)
a.setZeroVelocity()

...
```

RBC Protocol: Simulation(2)

```
for itime in u.cycle():

    if not RESTART:
        # gently increase excluded volume
        if itime <= GROWTIME:
            tscale =
0.5*(1.+itime/(1.*GROWTIME))
            if myid==0 and itime%100==0:
                print 'Rescaling interaction to',
tscale
            a.scaleVdwParameters(tscale)

        # increase plasma-RBC coupling
        elif itime == int(1.5*GROWTIME):
            f.setFreeze(False)
            a.setGamma(gammaT=1.e-3,
gammaR=1.e-3)
            a.setZeroVelocity()

         #RBC free to move with right
coupling
        elif itime == int(2.*GROWTIME):
            a.setGamma(gammaT=1.e-2,
gammaR=1.e-2)
            a.setZeroVelocity()

        # allow the drug to move
        elif itime == int(2.5*GROWTIME):
            c.setFreeze(False)

    # BC: stripe of given drug density
    c.setDensityStripe('x', nx, C0)

    u.animate()
```

RBC visualization

In the DIRDATA_... folder relative to RBCs, the subfolder VTK contains data for cells. In paraview they can be visualize as simple points or by using RBC-like glyphs:

- Load *pvtu files in paraview
- Load the glyph RBC.vtk (provided in the github repo)
- Use the filter: "Glyph with custom source" and select the *.pvtu for the input and RBC.vtk for Glyph Source
- Set:
- Scale Factor => 1
- Orientation => "orientation"
- Glyph Mode => "All points" (if ok interactively!)

For batch mode, learn how to use paraview in python, let's talk.