COSMOS 2022, Computational Lab 04

Molecular Dynamics (MD)

MD is a very powerful and commonly used research tool to compute trajectories of collections of particles moving under the laws of *classical* mechanics. MD is employed, for example, by astrophysicists to study galaxy formation (the 'particles' are stars!), and by biologists to study protein folding (the 'particles' are the molecules comprising the protein).

MD is based on two equations. First, if a particle has a velocity v then in a short time dt it moves a distance

$$dx = v\,dt$$

This is an intuitively reasonable statement based on what we mean by 'velocity'. It can also be seen from the physics formula $v = dx/dt$. Second, if the particle experiences a total force F then in a short time dt the velocity changes by

$$dv = \frac{F}{m}\,dt$$

This second equation is not so intuitive, but is reasonable in the following sense: If a particle has a big mass m we expect it is harder to affect its motion. This is reflected in the equation by the fact that m is in the denominator. A more rigorous derivation is to put together the two equations $a = dv/dt$ and $F = m\,a$.

Usually the force F depends on the particle's location and velocity: $F = F(x, v)$.

A key assumption of these equations is that v and F are close to being *constant* throughout the interval dt. This is why dt needs to be small.

The way *all* molecular dynamics codes work is you put these two equations into a loop. Each step in the loop allows a short time dt to pass, and if you do N steps, then a long time $t = N\,dt$ passes. You get the path of the particle during this time.

If your particle moves in 3D then you have three equations for x, y, z and three equations for v_x, v_y, v_z.

If you have many particles you have positions and velocities for each one (usually stored in an array).

Molecular Dynamics for object thrown vertically upwards.

A simple example is a mass m moving under the force of gravity $F = mg$ where $g = -9.8$ m/s^2 is the acceleration of gravity. Here is a code which does molecular dynamics for this problem:

```
#include <stdio.h>
#include <math.h>
int main()
{
     double y,v,t,dt,F,m,g=-9.8;
     int j,N;
     FILE * fileout;
     fileout=fopen("Hagrid","w");
     printf("\nEnter mass m:   ");
     scanf("%lf",&m);
     printf("\nEnter starting height y and velocity v:   ");
     scanf("%lf %lf",&y,&v);
     printf("\nEnter time step dt and number of steps N:   ");
     scanf("%lf %i",&dt,&N);
     t=0.;
     fprintf(fileout,"\n   %12.6lf %12.6lf",t,y);
     for (j=0; j<N; j=j+1)
     {
          t=t+dt;
          y=y+v*dt;
          F=m*g;
          v=v+(F/m)*dt;
          fprintf(fileout,"\n   %12.6lf %12.6lf",t,y);
      }
      fclose(fileout);
      return 0;
}
```

• Note a new features in this code: We need to write N positions out to specify the trajectory $x(t)$. This is too much to dump to the screen, so we instead print to a file. This involves:

Declaring a filename: FILE * fileout;

Opening the file: fileout=fopen("Hagrid","w");

You can use whatever creative name you like for the filename. Go wild!

Using 'fprintf' instead of 'printf': fprintf(fileout,"\n %12.6lf %12.6lf",t,y);

Closing the file: fclose(fileout);

• Note also that m cancels out in this code. This only happens for the force of gravity! Indeed, it is a big mystery why the m in $F = ma$ is identical to the m in $F = mg$!

• Run your code for starting height $y = 4$ m, starting $v = 12$ m/s, time step $dt = 0.01$ s and $N = 500$ steps.

• How long does it take for the mass to reach its maximum height?

• How long does it take for the mass to hit the ground ($y = 0$)?

• If you have taken a physics course, you should be able to use some equations you learned there to check your code.

• This code is awkwardly written in that the particle keeps moving even after it crashed into the ground! That is, y is allowed to go negative.

V2: Molecular Dynamics for object thrown vertically upwards.

A much better way to write the code is to use a do-while loop instead of a for loop.

```
#include <stdio.h>
#include <math.h>
int main()
{
     double y,v,t,dt,F,m,g=-9.8;
     int j;
     FILE * fileout;
     fileout=fopen("Hagrid2","w");
     printf("\nEnter mass m:   ");
     scanf("%lf",&m);
     printf("\nEnter starting height y and velocity v:   ");
     scanf("%lf %lf",&y,&v);
     printf("\nEnter time step dt:   ");
     scanf("%lf",&dt);
     t=0.;
     fprintf(fileout,"\n   %12.6lf %12.6lf",t,y);
     do
     {
          t=t+dt;
          y=y+v*dt;
          F=m*g;
          v=v+(F/m)*dt;
          fprintf(fileout,"\n   %12.6lf %12.6lf",t,y);
      }while(y>0);
      fclose(fileout);
      return 0;
}
```

• Is the syntax/logic pretty sensible? You **do** the steps in the loop **while** the condition $y > 0$ holds.

• do-while loops are good when you know the condition for when the loop will stop, but do not know how many steps N it will take.

• Run your code for starting height $y = 4$ m, starting $v = 12$ m/s, time step $dt = 0.01$ s. We do not need to specify N here! Compare the output to the previous version. (Do you see where the output is going this time?)

(optional) V3: Molecular Dynamics for object thrown vertically upwards.

Add an air resistance force $-bv$ to your code so that the total force is $F = mg - bv$. Run your code with initial $y = 4$ m/s, $v = 12$ m/s, $dt = 0.01$, and $b = 0.1$, $m = 0.5$ kg. Notice m matters now (because a force other than gravity is present)!

- How does the maximum height compare to your previous code (no air resistance)?
- How does the time to reach the ground compare?

Plotting object thrown vertically upwards.

Make a plot of y vs t for [07]. Modify your code to print out the pair t, v (just change one character!) and then also make a plot of v vs t for [07]. What are the shapes you find? Can you 'correlate' the features in the two plots? For example, when y reaches its maximum value what is v doing?

(optional) Connecting to a physics course

If you have had a physics course you may have seen the equations.

$$y(t) = y(0) + v_y(0)\,t + \frac{1}{2}g\,t^2$$
$$v_y(t) = v_y(0) + g\,t$$

Do your plots/results agree with these equations?

(optional) (More!) Connecting to a physics course

If you have had a physics course you may have been taught about conservation of energy:

$$E = \frac{1}{2}mv^2 + mgy$$

Use the fact that E is constant (in the absence of air resistance!) to compute the maximum height. (Find E at $t = 0$ and then use the fact that $v = 0$ when you are at maximum height.)

(optional) Terminal velocity!

Run your **optional** code for the problem of dropping an object off the Empire State building. Use initial values $y = 400$ (the height of the building), $v = 0$ (release from rest), $dt = 0.01$, and $m = 0.5$. First do a MD simulation for $b = 0.0$ (no air resistance). How long does it take for the object to hit the ground? Then run with $b = 0.1$. Now how long does it take to hit the ground? Plot $y(t)$ for both cases. It is best to plot them on the same graph so you can easily compare! Describe the difference in the shapes of $y(t)$ for $b = 0.0$ and $b = 0.1$.

Finally, do the same comparison of $b = 0.0$ and $b = 0.1$ for $v(t)$. Can you intepret what you are seeing. Hint: Have you ever heard the phrase 'terminal velocity'?