1 N-body problems

A well known n-body problem involves calculation of gravity forces between N object, where the force can be between object i and j is determined by an equation

$$f_{ij} = G \frac{m_i m_j}{\| r_{ij} \|^2} \cdot \frac{r_{ij}}{\| r_{ij} \|}$$

,where the f_{ij} is the gravity on object i caused by object j. However, there is a problem when the divisor is close to zero. To fix the error, many studies suggested to and a small constant into the divisor to get smoothing effect [Aarseth 2003, Dyer and Ip 1993].

$$a_i \approx G \sum_{1 \le j \le N} \frac{m_j r_{ij}}{\left(\| r_{ij} \|^2 + \varepsilon^2 \right)^{1.5}}$$

A python code for calculating all forces in the system can be simple as below.

```
import numpy as np
import matplotlib.pyplot as plt


N = 100
s = np.random.rand(N,2)*2-1
r = np.zeros((2,N,N))*2-1
v = np.zeros((N,2))

m=1.0
dt=0.01
G=0.1
epsilon=1e-3
while True :
    for i in xrange (N):
        for j in xrange (N):
            r[:,i,j] = s[j,:] - s[i,:]

    c = (r[0,:,:]**2 + r[1,:,:]**2 + epsilon**2)**(-1.5)
    a = r*c
    np.fill_diagonal(a[0,:,:], 0)
    np.fill_diagonal(a[1,:,:], 0)
    ai=np.transpose( G * m * np.sum(a,axis=2) )
    v = v + ai*dt

    s = s + v*dt

    plt.plot(s[:,0],s[:,1],'.r')
    plt.hold(False)
    plt.axis([-20,20,-20,20])
    plt.draw()
```