Subject: Avoiding a FOR loop in calculation of SPH potential energy
Posted by cody on Tue, 23 Jun 2009 06:48:00 GMT
View Forum Message <> Reply to Message

i have an SPH simulation with snapshots in time that are represented as structures in idl. the structures have mass, density, position etc. for every particle in the simulation. i'm trying to get a vector that represents the total potential energy at each snapshot. i have an outer FOR loop that loops through each snapshot file and this part is fine, i don't really want to change that. but i have an inner loop for calculating the PE that must calculate the total potential energy felt by each particle from every other particle without double counting. to do that, i just loop over a dummy variable j from 0 to the maximum particle ident -1 and make a sub list of particles where ident > j. this way, i'm not double counting and it's not n^2 time. but it's still way too slow for a 100k particle sim.

now i'm fairly certain there's a way to speed this up considerably using IDL's strengths, but i'm a c++ programmer, so that's the way i think. can someone help me out?

here's my code as it stands now:

```
PRO exportenergy, enditer=enditer, deliter=deliter, root=root


;first get the total number of particles
startiter = 1000
n = (enditer - startiter)/deliter
m=0
G=3.93935d-7 ;SPH units

arr = DBLARR(5, n)
for i=0L,(n-1) do begin
  iter = i*deliter + startiter
  filename = root + '.' + STRING(iter,format='(I0)')
  readsdf,filename,s
  arr[0,i] = sdf_getdouble(filename,"tpos")
  arr[1,i] = 0.5*total(s.mass*s.vx^2)
  arr[2,i] = 0.5*total(s.mass*s.vy^2)
  arr[3,i] = 0.5*total(s.mass*s.vz^2)

  ;PE part starts here
  pn = max(s.ident)
  PE = 0.0
  for j=0L,pn-1 do begin
    sub = where(s.ident gt j)
    PE = PE + G*s[j].mass*total(s[sub].mass/sqrt((s[j].x-s[sub].x)^2+
```

```
(s[j].y-s[sub].y)^2+(s[j].z-s[sub].z)^2))
      print,'just computed ' + string(j,format='(I0)') + '  PE=' +
string(PE)
   endfor
   arr[4,i] = PE
endfor
```