Subject: Re: Astronomys` Sixth Neighbour Needs Help
Posted by robert.dimeo on Fri, 25 Jul 2003 13:23:42 GMT
View Forum Message <> Reply to Message

touser2001@yahoo.com (astronomer) wrote in message
news:<c2959ed8.0307240506.1136eacd@posting.google.com>...

> Basically, the program reads a 2-column n-line file (n is number of
> stars in field, the columns are the position of the stars, Right
> Ascencion and Declination which are similar to latitude and
> longitude). My objective is to obtain, for each star, the distance to
> its 6th neighbour. Simply, Right ascencsion is x and Declination is y
> on a xy grid.
>

Hi Bruno,

I tried to streamline your code and eliminate loops using as many
array operations as I could. In IDL this can significantly increase
the speed over using loops. In the code that I wrote below I used a
procedure called WHERETOMULTI.PRO that converts 1-dim indices to 2-dim
indices. You can do a search on this newsgroup for that program.
Alternatively you can wait until IDL 6.0 comes out and use the nifty
ARRAY_INDICES routine.

The following function returns the distance for the 6th nearest
neighbor given x-y coordinates in the order in which the coordinates
are given. So d6[i] is the distance to the 6th nearest neighbor for
the star located at coordinates (x[i],y[i]).

```
FUNCTION CALCULATE_6TH_NEIGHBOR,x,y
; Calculate all of the distances
n = n_elements(x)
u = 1+bytarr(n)
dx = u#x-x#u
dy = u#y-y#u
d = sqrt(dx^2+dy^2)
dsort = sort(d)
wheretomulti,d, dsort, col, row
; Now get the sixth neighbor
d6 = fltarr(n)
for i = 0L,n-1 do begin
 index = where(col eq i)
 d6[i] = d[col[index[6]],row[index[6]]]
endfor
return,d6
END
```

On my 1.7 GHz PC with 1 GB of ram I processed 1000 x-y pairs in 15.9 seconds. I should mention that when I tried to process 10000 x-y pairs my machine was unable to allocate enough memory to create the arrays. So this may not have solved your problem but it might lead you in the right direction for how to remove loops to gain speed.

Hope this helps.

Rob
