Subject: Re: Distance calculation for lots of stars
Posted by wlandsman on Mon, 13 Jul 2015 18:36:30 GMT
View Forum Message <> Reply to Message

A couple of comments:

You are not using the correct formula for the distance on a sphere. (Near the pole, stars can have very different right ascensions but be separated by less than an arc second.)

If stars are separated by more than 1 arc second in declination, then they must be more than 1 arc second apart. So a first step is to just search in declination, and you only need the full distance computation when the declination differs by less than 1 arc second.

Jeremy Bailin's program matchall_sph.pro uses many such tricks to speed up the processing. It is available in his tar file jib-1.2.tgz available from
http://www.simulated-galaxies.ua.edu/jbiu/ --Wayne

On Monday, July 13, 2015 at 1:45:09 PM UTC-4, Matthew wrote:
> Hello,
>
> I have been tasked with finding stars within 1 arcsecond of each other through multiple epochs of data over a few years.
>
> I have developed a program to do so but it takes quite a long time to complete due to for loops and a very large data set.
>
> I would like to optimize the part of my code that loops through each star individually (categorized by epoch) and compares it against all other stars in the data set. This is clearly the step that is taking the longest and if I optimize this step then the entire process should speed up exponentially. All other for looping that is done is done over much smaller sets of data so I'm not too worried about those.
>
> This step is a very basic distance calculation and I'm wondering if there is a quicker way to do it. So far that part of my code looks like this:
>
> for f = 0, sectioncount-1 do begin
> ;Sectioncount is the individual stars in a certain epoch to search for
>
> StarDist = sqrt((arcALLRA-arcRA(f))^2+(arcAllDEC-arcDEC(f))^2)
>
> multistars = where(Stardist lt 1, starcount)
> ;Finding the location of the stars that are within one arcsecond after doing the distance calculation
> ...
>
> Since any iteration in the for loop doesn't affect the next, I'd be willing to bet that there's a faster way to do it. Is there any way to speed up the distance computation for all stars in the 'individual star' data set?

>
> I have looked at David Fanning's post on the matter and tried to use Match_2d from JD Smith. Every time I do, I encounter an error and have since abandoned the idea.
>
> Any help is greatly appreciated,
>
> Matthew
