Subject: Re: Strange floating-point precision behavior
Posted by James Kuyper on Mon, 10 Feb 2003 20:44:59 GMT
View Forum Message <> Reply to Message

Tim Lloyd wrote:
>
> I have written a routine that converts Earth-Centered Inertial
> coordinates in x/y/z to geodetic latitude/longitude/altitude using the
> WGS84 standard. I have one issue, however, that I believe is
> affecting my calculations of altitude so that they are accurate only
> to 1-meter resolution. I am defining the ECI coordinates as
> double-precision:
>
> IDL> boulder={x:-1283388.8693d0, $
> y:-4713016.9053d0, $
> z:4090191.0471d0} ;Boulder, CO, GPS station

Are you sure those are ECI coordinates? Interpreted as ECR
(Earth-Centered Rotating) coordinates, they correspond pretty closely to
Boulder CO. Interpreted as ECI coordinates, you'd need a fourth value,
the precise time at which the conversion from ECI to ECR should occur.
ECI and ECR coordinates match only once each day, so it would be quite a
coincidence if those ECI coordinates happened to match the ECR
coordinates for Boulder.

> and yet IDL seems to be storing the data incorrectly:
>
> IDL> print,boulder,format='(3f20.10)'
> -1283388.8692999999 -4713016.9052999998 4090191.0471000001
...
> What am I doing wrong? I am fairly certain that this behavior is
> responsible for my calculations yielding 1674.6658 m as the altitude
> of the Boulder GPS station, and not 1674.7428 m (the actual altitude).

No, that isn't the cause of your problem. Floating point roundoff has
introduced errors of only about 10^-10 meters into your calculations;
that can't be the cause of a 0.123 meter error in the altitude. I have
access to a C routine which performs this same conversion, and it
produces the same result as your routine. Is it possible that it's not
the routine that's at fault, but the data?