
Subject: Re: RANDOMU bug (and HTML help)
Posted by [thompson](#) on Tue, 23 Feb 1999 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

ajschmitt@my-dejanews.com writes:

> In article <21FEB199900213357@stars.gsfc.nasa.gov>,
> landsman@stars.gsfc.nasa.gov (Wayne Landsman) wrote:
>> Versions 5.1.1 and 5.2 of IDL have a bug in the RANDOMU (and RANDOMN) function
>> such that the SEED variable is initialized to the same value at the start of
>> each session, rather than being initialized by the system clock (see example
>> below). I believe that so long as one stays within one IDL session
>> that this causes no problems, but if one is, say, combining Monte Carlo
>> simulations from different IDL sessions, then the results will be decidedly
>> unrandom.

>>
>> RSI knows about this problem and say that they are giving it very high
>> priority for a fix.

>>
> This is not the worst of it. For some of us, it is important to be able
> to have the same seed so that the same "random" sequence is produced.
> However, RSI quietly changed the manner in which the SEED variable
> interacts with the RANDOM functions between version 5.0 & 5.1.

> Consider the following behavior from IDL v 5..0.3:
> IDL> seed = 2 & print, randomu(seed, 3)
> 0.342299 0.402381 0.307838
> ...doing this multiple times will always give the same result.

> However, in IDL v.5..1 & later,. using this several times in a row
> produces different results each time.:
> IDL> seed = 2 & print, randomu(seed, 3)
> 0.0594004 0.982075 0.358593
> IDL> seed = 2 & print, randomu(seed, 3)
> 0.831999 0.303037 0.506712

> ...etcetera. It turns out that you now have to specify a NEGATIVE seed
> in order for it to have any influence on the generated sequence:

> seed = -2 & print, randomu(seed, 3)
> 0.342299 0.402381 0.307838
> seed = -2 & print, randomu(seed, 3)
> 0.342299 0.402381 0.307838

(rest deleted)

That's interesting. When I try this in IDL/v5.1.1, I get the same result over and over again, even with positive seeds.

```
IDL> seed = 2 & print,randomu(seed,3)
  0.342299  0.402381  0.307838
IDL> seed = 2 & print,randomu(seed,3)
  0.342299  0.402381  0.307838
IDL> print,!version
{ alpha OSF unix 5.1.1 Jul 20 1998}
```

The online help, though, still states that one is supposed to put in a negative number to re-use a seed. In fact, the online help in version 5.1.1 is rather misleading in that it implies that the seed is a scalar value, whereas it's actually returned as a 36-element array just like in 5.2. The 5.2 online help does away with any mention of needing to put in negative values.

It looks like the business of needing to put in a negative number to force seed initialization is restricted to version 5.1.0. Maybe fixing that problem is what introduced the other problem?

Here's another weird behavior of RANDOMU. If you call IDL with an undefined seed, it's supposed to initialize the seed for you. Thus, when you type in

```
IDL> print,randomu(seed,3)
  0.653919  0.0668422  0.722660
```

before seed is defined to anything, it still works. Naively, I expected that if one deleted the seed variable, that would force IDL to re-initialize the seed, based on the system time or whatever. However, if my next command is

```
IDL> delvar,seed & print,randomu(seed,3)
```

I don't get any random numbers at all. To start getting random numbers again, I have to use a completely new variable name for the seed, e.g.

```
IDL> print,randomu(seed2,3)
  0.671149  0.383416  0.631635
```

I don't know if this is properly speaking a bug or not, but it's certainly weird. As far as I can tell, this behavior is seen in all versions of IDL.

Bill Thompson
