Subject: bug in IDL's hanning() window-generating function
Posted by bennetsc on Wed, 01 Aug 2001 08:16:38 GMT
View Forum Message <> Reply to Message

Perhaps this has been discussed or reported before, but if so, RSI has not yet fixed it.

I just noticed that IDL's hanning() has a bug described by Harris (1978) as common in software packages. The bug is easy to look for. To check, generate a window of even length. Then look at its coefficients. The array should *not* be exactly symmetrical. It should have only one highest coefficient next to, and to the right of, the "center" of the array, noting that the center of the time (or space) period covered by the discrete window actually falls at the largest window coefficient, which is immediately to the right of the "center" of the array, because of the even number of coefficients. IDL's hanning() produces a symmetric window with two identical coefficients at the center and zeros at the ends, which is incorrect. A Hann window of even length should only be zero at the left (i.e. starting) end, whereas the other end should be equal to the first coefficient in from the zero. (A Hann window of odd length should also be zero at the start, but not at the end. See further below.)

For example, "print, hanning(6)" gives

0.00000 0.345492 0.904509 0.904508 0.345491 0.00000

whereas the correct sequence is

0.00000 0.250000 0.750000 1.00000 0.750000 0.250000

Next, generate a window of odd length. An odd-length window should have its largest coefficient at its central element, but the next coefficient should be identical to it, followed by decreasing values out to the end of the window that are the mirror image of values going the other direction from the central element. This results in the last coefficient of the window still matching the second coefficient of the window. The duplication of the highest coefficient in the middle of the window, with the duplicated value being less than 1, seems to me to account for the time (or space) shift in sampling a given periodic function that one finds when using 2n + 1 samples instead of 2n samples. In other words, the center of the time (or space) period covered by the discrete window falls halfway between the two largest data window array elements. Note that this means that the single highest coefficient of a typical window of even length will be 1, but the two highest coefficients of a window of odd length will be equal and somewhat less than 1; i.e. the value 1 would fall halfway between the two coefficients actually present at the discrete sampling interval.

For example, "print, hanning(5)" gives

0.00000 0.500000 1.00000 0.500000 0.00000

but the correct sequence should be

0.00000 0.345492 0.904509 0.904508 0.345491

Note that the third and fourth coefficients above are "identical"
within the limits of precision and the default format used in the
print statement, as are the second and fifth. Also, note that the
same kind of error is produced regardless of the value of ALPHA coded
(default is .5, which yields a Hann window) in the call to hanning(),
excluding ALPHA=1., of course.

In any case, I think the point Harris made is that a discrete
sampling of a window function should not taper to the same value at
the end that it has at the beginning because to do so would include
the first sample of the *next* period (windowed segment.) So IDL's
hanning() gets it wrong for both even- and odd-length windows. :-(

Ref.:
Fredric J. Harris, On the Use of Windows for Harmonic Analysis
with the Discrete Fourier Transform, Proc. IEEE, vol. 66
no. 1, January 1978

Scott Bennett, Comm. ASMELG, CFIAG
College of Oceanic and Atmospheric
Sciences
Oregon State University
Corvallis, Oregon 97331
********** **********
* Internet: sbennett at oce.orst.edu *
---------- ---------
* "Lay then the axe to the root, and teach governments humanity. *
* It is their sanguinary punishments which corrupt mankind." *
* -- _The_Rights_of_Man_ by Tom Paine (1791.) *
********** **********