
Subject: Convolution of Stick Spectra

Posted by [tclement](#) on Sat, 09 Sep 2000 22:28:49 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi all...

In my on-going effort to speed up the code in our lab, I have another 'challenge' for you (I have to put 'challenge' in quotes because it seems that no matter what I ask, someone knows the answer off (or at least nearly off) the top of their head!)

We have a situation where we need to convolute (with energy dependent gaussians) a number of stick spectra on a well-defined energy axis. The stick spectra are read in from another program as a 2-dimensional array, using `ddread`. The `[0,*]` elements are the energies of the sticks, and the `[1,*]` values are the intensities. These have no inherent spacing, they are just calculated intensities at whatever energy the calculation returns.

We want to convolute these with gaussians on a regular energy axis, so we declare our energy array from 0 to 4, as shown below.

The only way I've found to do the convolution, since the x axes don't match on the two arrays, is to do the following (slow) for loop:

```
convoluted = fltarr( 2, 2048 )
convoluted[0,*] = findgen( 2048 ) / 2047. * 4.
```

```
;; Let's fake a stick spectrum, we usually have at least this many elements
stick = abs(randomn( systime(1), 2, 5000 ))
stick[0,*] = stick[0,*] * 4.
stick[1,*] = stick[1,*] * 1000.
```

```
for i=0L, n_elements( stick ) / 2 -1 do $
  convoluted[1,*] = stick[1,i]*exp(-((convoluted[0,*] - stick[0,i])^2)/ $
    (((.12*sqrt(stick[0,i]/1000))/1.6651)*1000)^2) $
    + convoluted[1,*]
```

So the question becomes, is there any way to speed this up? I thought of using a larger convoluted array with multiple dimensions and using `total()` somehow, but I couldn't think of how to do that.

Thanks in advance (and in retrospect) for all the help!
Todd
