

VI. Primer on Lorentzian and Gaussian (Bruker) Multipliers [MathCAD import]

by cg fry: revised 28-Oct-98

This discussion should illustrate the very different effects of Gaussian and Lorentzian (or Exponential) multipliers on resulting NMR spectra. Lorentzian multipliers are used to decrease noise in spectra, but at the price of lower resolution. Gaussian multipliers, on the other hand, are only used to enhance the apparent resolution of a spectrum, and should be used with caution, as noise spikes can be transformed into what appear to be peaks under Gaussian multiplication.

This document was made using MathCAD. All calculations are shown, with resulting plots. A "live" MathCAD document allows a student to interactively change parameters (LB and GB here) to see the effect of the changes directly. Experimentation is often the best teacher, so give it a try.

First, we set-up the time domain points:

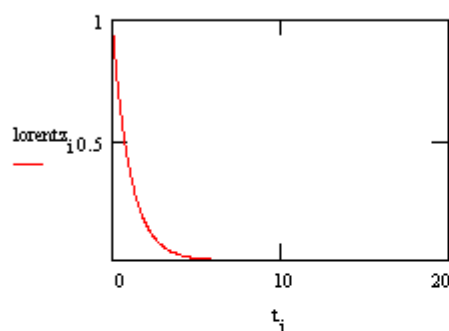
$npts := 1024$ $i := 0..npts$ i is the counter index

$AQ := 20 \text{ s}$ $t_1 := AQ \cdot \frac{i}{npts}$ $DW := t_1$ $SW := \frac{1}{DW}$

$DW = 0.02 \text{ s}$ $SW = 51.2 \text{ Hz}$

Now we can calculate a Lorentzian - or exponential - weighting function:

$LB := .3$ $lorentz_1 := \exp(-t_1 \cdot LB \cdot \pi)$



Matched filter conditions use $LB = 1 / (\pi T_2)$.

A different description is: LB equals the full-width at half-maximum of the peak without weighting functions. Exponential multipliers improve S/N, but increase the linewidths (by LB Hertz).

Good values for matched filters are $LB = 0.2$ on the WP's

$LB = 0.4$ on the AM's

$LB = 1-3$ for ^{13}C data

Keep in mind that LB on PCNMR4Windows is the constant entered in the box directly in line with Lorentzian button. GB used later is the constant entered in the box in line with the Gaussian button.

Now generate a simulated FID of a poorly resolved overlapping pair of doublets:

$f1a := 20$ Hz $f2a := 20.5$ - f## is the frequency of the peaks in Hz
 $f1b := 23$ $f2b := 23.3$
 $w1 := .3$ Hz $w2 := .35$ - w# is the full width at half-max of the peaks

$$fid1_i := \left(\exp(-2 \cdot \pi \cdot t_i \cdot f1a \cdot j) + \exp(-2 \cdot \pi \cdot t_i \cdot f1b \cdot j) \right) \cdot \exp\left[-\left(t_i \cdot w1 \cdot \pi\right)\right]$$

$$j := \text{sqrt}(-1)$$

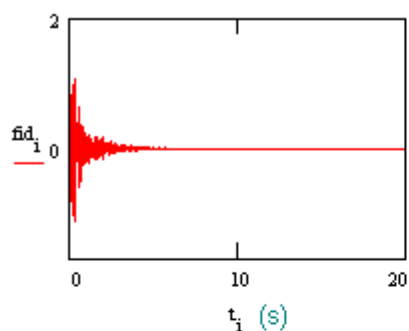
$$fid2_i := \left(\exp(-2 \cdot \pi \cdot t_i \cdot f2a \cdot j) + \exp(-2 \cdot \pi \cdot t_i \cdot f2b \cdot j) \right) \cdot \exp\left[-\left(t_i \cdot w2 \cdot \pi\right)\right]$$

$$fid_i := \frac{fid1_i + fid2_i}{2}$$

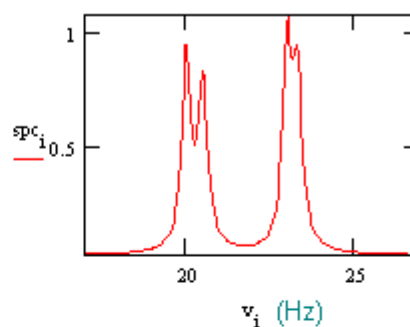
Calculate the Fourier variable, ν , which just the frequency in Hz, and take the Fourier transform:

$$\nu_i := \frac{i}{npts \cdot DW} \quad spc_i := \text{Re}(\text{cfft}(fid))$$

The FID appears as



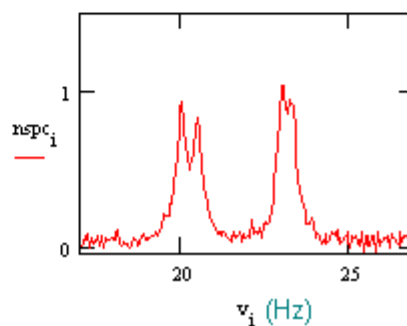
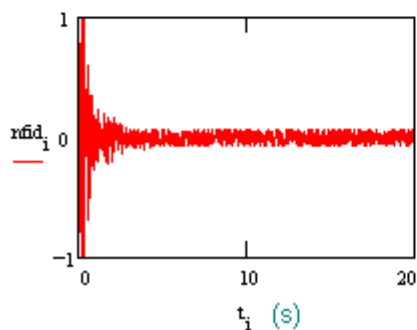
The spectrum is



Add some noise using the random number function, rnd :

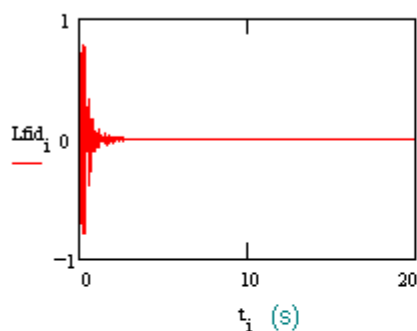
$$nfid_i := fid_i + \frac{\text{rnd}(1) - 0.5}{7}$$

$$nspc_i := \text{cfft}(nfid)$$

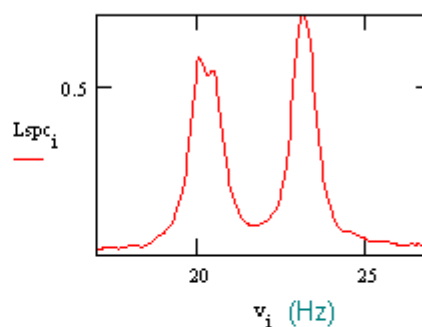


Now we apply the Lorentzian weighting function:

$$\text{Lf}id_1 := \text{nfid}_1 \cdot \text{lorentz}_1$$



$$\text{Lspc} := \text{fft}(\text{Lf}id)$$



We see that the noise is decreased significantly, but at the expense of loss of resolution.

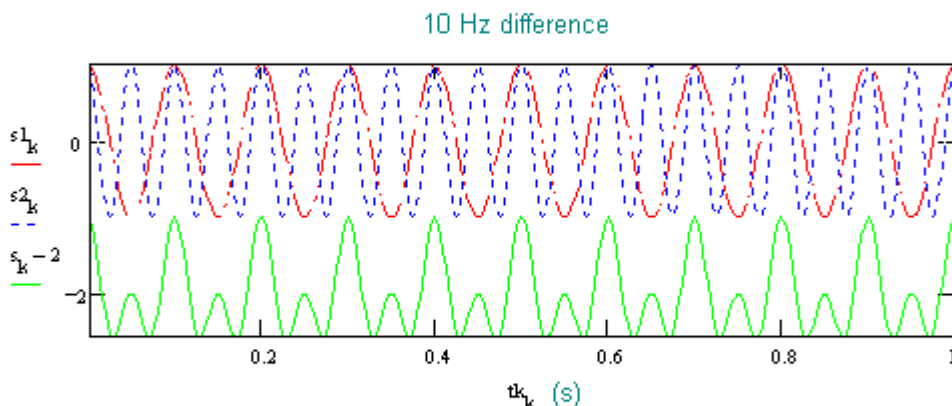
Now look at a Gaussian function. This function is useful for artificially enhancing the apparent resolution of a spectrum, which may be useful in the simulated spectrum shown above. The peaks at 23 and 23.3 Hz are barely resolvable (not at all after Lorentzian multiplication!).

A Gaussian multiply works by increasing the portion of the FID where the best resolution is occurring in the FID. This is the furthest time portion of the FID where there is still signal: in the noisy FID shown at the bottom of page 2, this time is ~ 4 s. Increasing the long-time component increases the apparent resolution, because the further out in the FID we can see signal, the better the resolution. The follow graphs illustrate why this is so:

Take two cos signals separated by 10 Hz:

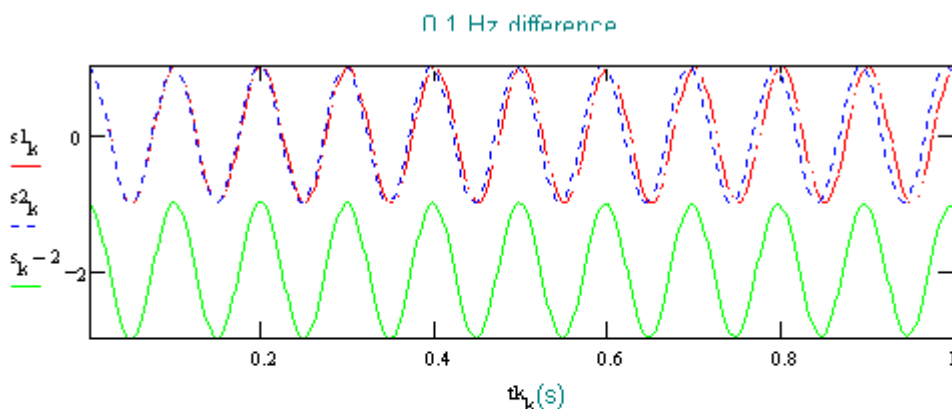
$$k := 1 \dots 400 \quad tk_k := \frac{k}{400}$$

$$s1_k := \cos(2 \cdot \pi \cdot tk_k \cdot 10) \quad s2_k := \cos(2 \cdot \pi \cdot tk_k \cdot 20) \quad s_k := \frac{s1_k + s2_k}{2}$$



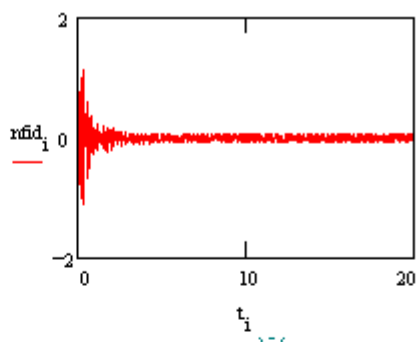
Clearly, very large difference are apparent by $1/10 = 0.1$ s into the FID. Now, however, let's look at two cos signals separated by 0.1 Hz:

$$s1_k := \cos(2 \cdot \pi \cdot tk_k \cdot 10) \quad s2_k := \cos(2 \cdot \pi \cdot tk_k \cdot 10.1) \quad s_k := \frac{s1_k + s2_k}{2}$$



In this case, almost no difference can be seen at 0.1 s. The key point is that smaller frequency separations can only be discerned further out in the FID. Resolution is therefore determined by having signal as far out in time as possible: the Gaussian multiplication therefore scales down the beginning of the FID and scales up the later times. Our main chore in properly using the function is to make sure to scale up signal instead of noise, but also not reduce the beginning of the FID, where the best S/N is, too much.

Noisy FID without any multiply function (from page 2).



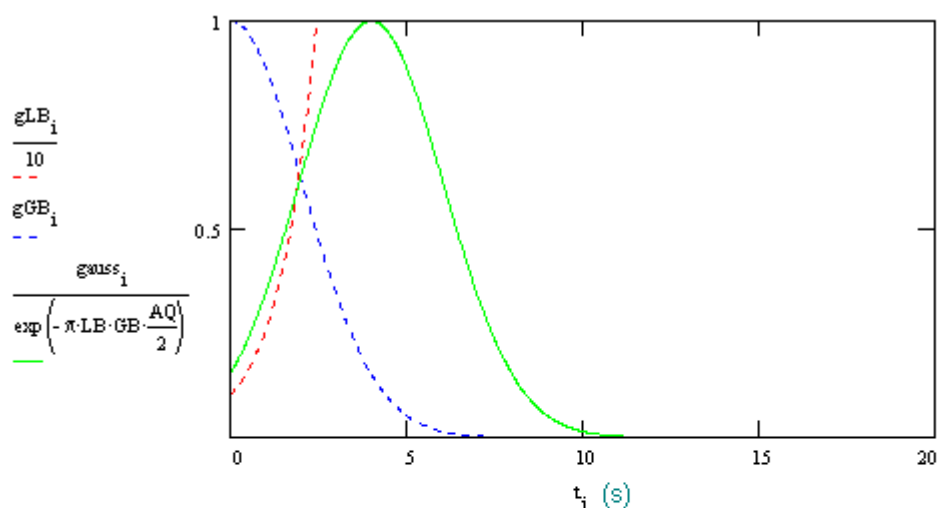
For Gaussian multiplication, choose a matched filter for LB, but always make it negative LB = -0.3 for the case introduced on page 2: shown again here for clarity.

$$LB := -0.3 \quad GB := 0.2 \quad gauss_1 := \exp \left[-t_1 \cdot LB \cdot \pi + \frac{(t_1)^2 \cdot LB \cdot \pi}{2 \cdot GB \cdot AQ} \right]$$

The function *gauss* is used in PCNMR4Windows and DISNMR. The Gaussian function is a multiplication of two terms involving LB and GB:

$$gLB_1 := \exp(-t_1 \cdot LB \cdot \pi) \quad \text{and} \quad gGB_1 := \exp \left[\frac{(t_1)^2 \cdot LB \cdot \pi}{2 \cdot GB \cdot AQ} \right]$$

so that now: $gauss_1 := gLB_1 \cdot gGB_1$ is the same expression as above.

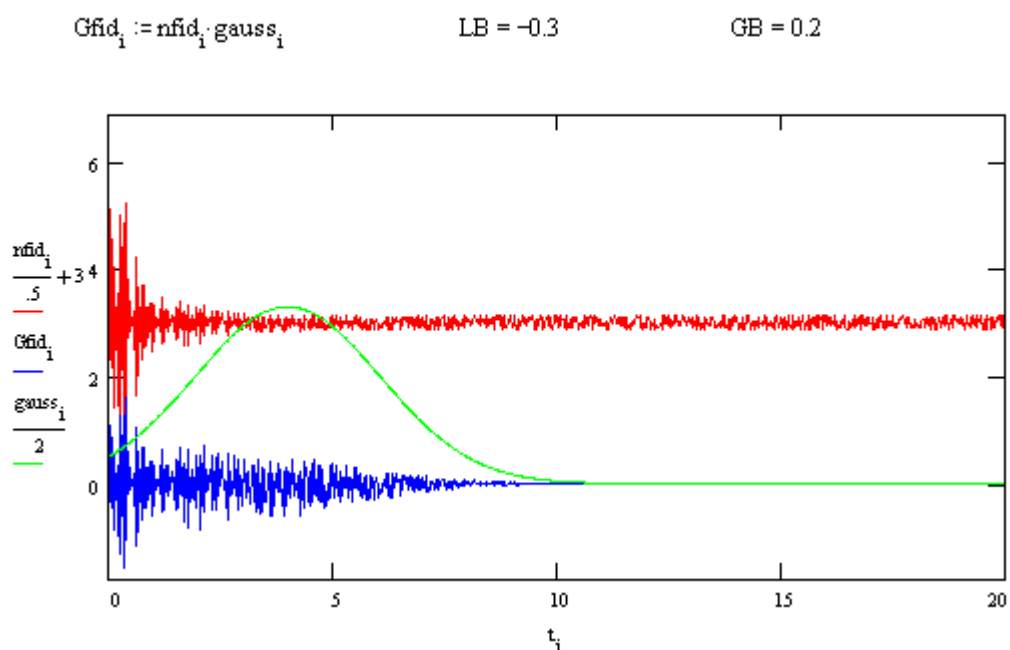


The dotted line is the first term, exponentially increasing because LB is negative. This term provides the enhancement of the later part of the FID. The dashed line is the second term, using both the LB and GB constants. This term is the Gaussian, t^2 , dependent term, and sets the maximum of the function along with the drop-off at large t . The combined terms are shown as the solid line, which can vary considerably, depending on the choice of LB and GB.

Experimenting with the function by changing LB and GB shows that GB sets the maximum of the Gaussian function as the fraction of AQ, and LB controls the width of the function. I.e. if we set GB = .5, the maximum will occur exactly halfway along the acquisition.

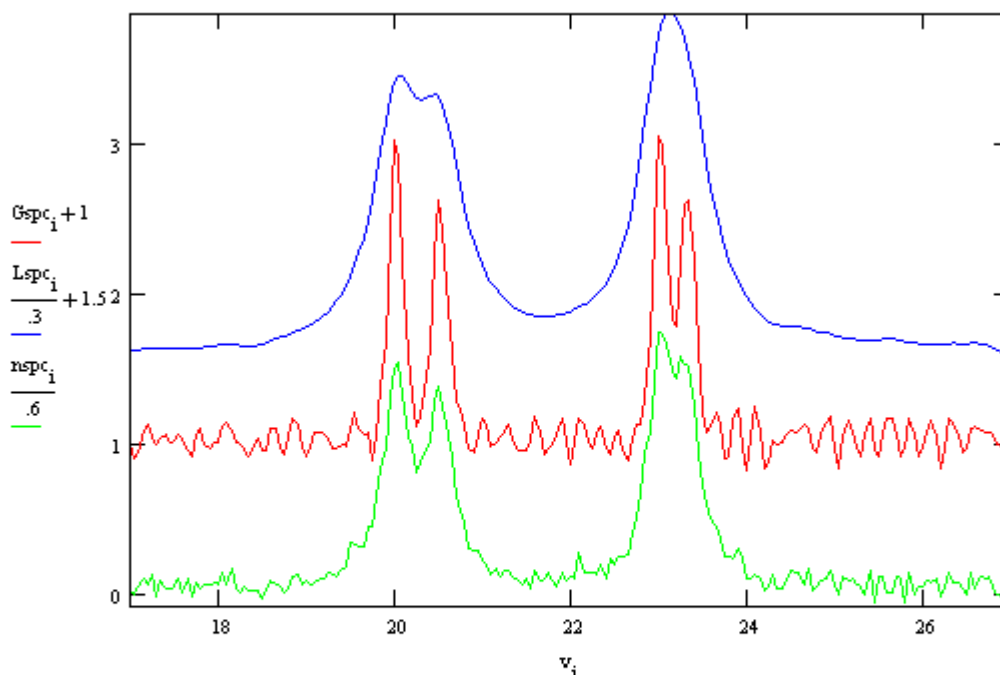
A good initial guess for GB is to find the first spot along the FID where signal is not apparent to the eye. In the FID above, this happens at ~ 4 s, or $4/20 = 0.2$ fraction of the acquisition: GB = 0.2.

Now watch the effect of the Gaussian function on the spectrum derived above:



As usual, the Fourier transformed spectra are more revealing:

$G_{\text{spc}} := \text{cff}(G_{\text{fid}})$



Clearly, the Gaussian multiplication has improved the apparent resolution, but at the cost of S/N. One should be very careful to not allow noise spikes to become apparent peaks when using Gaussian multipliers. Adjust the number dividing the `rnd` function on page 2 to see the effects of more noise. Notice also that making LB too large gives negative "ears" on the peaks, whereas GB too large scales up just the noise and not the signal by pushing the maximum of the function out too far.

As a final look, let's make a spectrum using poor values. First, use an LB that's too large negative, making negative "ears" on the peaks:

LB := -0.8 GB := 0.2

$$\text{gaussL}_i := \exp \left[-t_i \cdot \text{LB} \cdot \pi + \frac{(t_i)^2 \cdot \text{LB} \cdot \pi}{2 \cdot \text{GB} \cdot \text{AQ}} \right] \quad G_{\text{fidL}}_i := \text{nfid}_i \cdot \text{gaussL}_i$$

$G_{\text{spcL}} := \text{cff}(G_{\text{fidL}})$

Now make another spectrum with too large a GB, which will enhance not signal but noise:

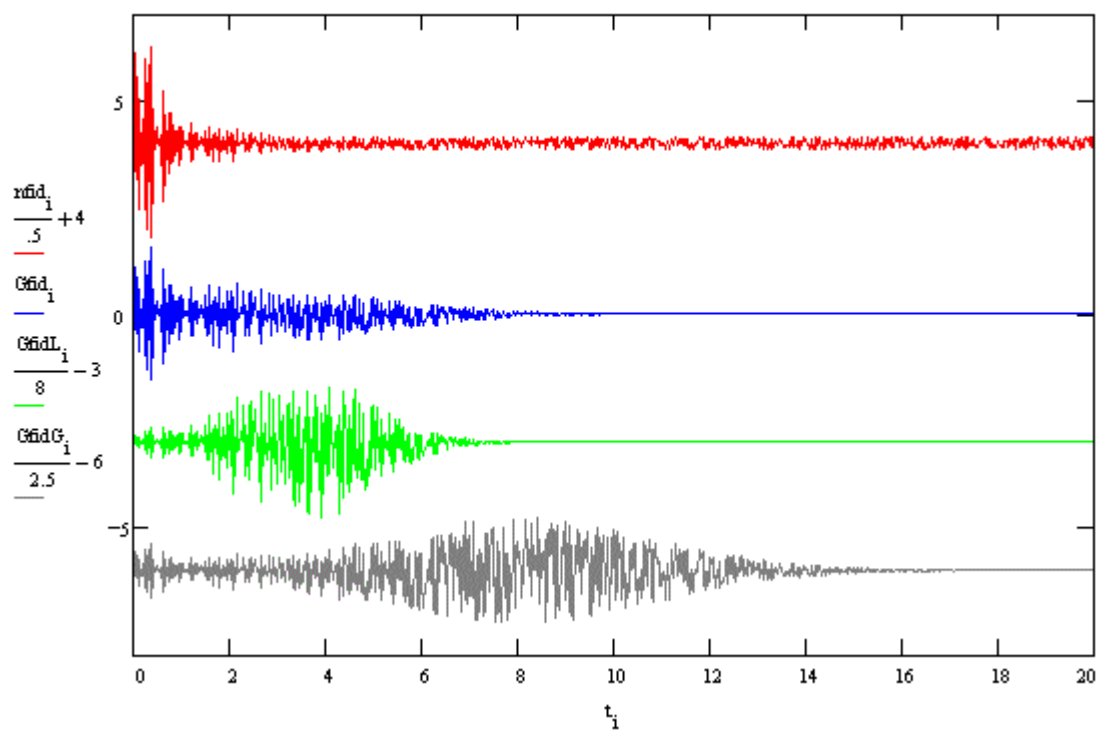
LB := -0.3 GB := 0.4

$$\text{gaussG}_i := \exp \left[-t_i \cdot \text{LB} \cdot \pi + \frac{(t_i)^2 \cdot \text{LB} \cdot \pi}{2 \cdot \text{GB} \cdot \text{AQ}} \right] \quad G_{\text{fidG}}_i := \text{nfid}_i \cdot \text{gaussG}_i$$

$G_{\text{spcG}} := \text{cff}(G_{\text{fidG}})$

11/12/93

FID



Spectra

