COHPROC, a coherent processing algorithm for the Chilbolton radars

Robin Hogan

Version 1.1, 7 December 2001

1 Introduction

The sensitivity of meteorological radars can be enhanced by spectral processing, which exploits the fact that the signal from meteorological targets often occupies a narrow velocity range in the Doppler spectrum, while the noise is spread evenly over the full unambiguous velocity range. Processing in the frequency domain also allows operations such as clutter suppression to be easily implemented. This document describes the coherent processing algorithm 'COHPROC', which has been designed to be used on all the Chilbolton radars. The CPUs of ordinary PCs are now fast enough to perform all the computations necessary for coherent processing (typically several thousand FFTs per second). By avoiding the use of dedicated DSP cards and implementing the algorithm entirely in C, COHPROC may be easily adapted and extended by non-specialist programmers.

In the next section the theoretical accuracies achievable by incoherent and coherent processing systems are calculated and compared. Then in section 3 the various stages of processing performed by the algorithm are described in detail. Finally in section 4 some of the shortcomings of the algorithm as it is currently implemented are outlined.

COHPROC was written by Gareth Davies, Robin Hogan and Darcy Ladd.

2 Achievable sensitivity of incoherent and coherent processing algorithms

2.1 Incoherent processing

An incoherent processing system makes use solely of the power measured by the receiver at each gate (in mW), and obtains the meteorological signal power by subtracting the 'noise' component, which can be due to both thermal and instrumental effects and is estimated from gates known to be free from cloud or precipitation. The sensitivity that can be achieved then depends on the magnitude of the natural fluctuations of the noise. This is illustrated in Fig. 1, which depicts the power (in linear units) measured by a hypothetical system, versus range. The power is the sum of the signal of the target (S) and the system noise (N). A signal is deemed to be present if the power exceeds some threshold level (shown by the dashed line). If the threshold is set too high then too many genuine target signals are rejected, but if it is set too low then the natural fluctuations in background noise can exceed the threshold and produce 'false alarms', which are apparent in processed radar data as spurious 'speckle noise' in regions that are known to be free from cloud and precipitation.

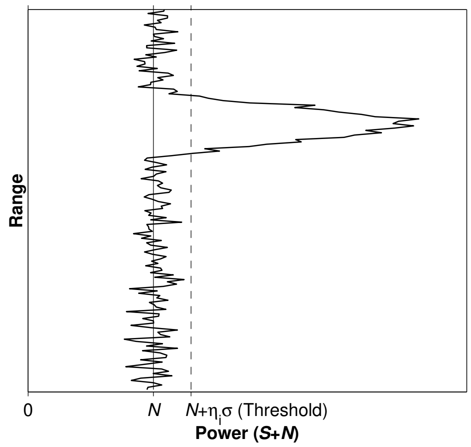

Figure 1: Schematic depicting received power versus range for a single ray of data (i.e. an average of a number of pulses).

By averaging a large number of pulses the fluctuations in N are reduced, which in turn enables the threshold to be lowered and smaller signals to be detected. The noise power measured from one pulse to the next is distributed inverse exponentially, and successive values are independent (i.e. the autocorrelation is zero). An inverse-exponential probability distribution has the property that the standard deviation is equal to the mean, so if we average m independent pulses then the standard deviation of the averaged noise power is given by

$$\sigma = \frac{N}{\sqrt{m}}. \quad (1)$$

Ordinarily the threshold would be set a fixed number (η_i)

of standard deviations above the mean noise level. A typical setting of $\eta_i = 3$ results in a false-alarm rate of around 0.13%, provided that the estimate of the noise level is reasonably accurate and that the background noise is free from any spurious interference effects. In practice most of the small number of false-alarm pixels will be surrounded by signal-free pixels in the full radar image, so can be rejected on that basis.

The minimum-detectable signal $S_{\rm min}$ produces a measured power exactly equal to the threshold value, so from (1) we obtain

$$S_{\rm min} = \eta_i \sigma = \frac{\eta_i N}{\sqrt{m}}. \qquad (2)$$

It is more convenient to express this as the minimum detectable signal-to-noise ratio in logarithmic units:

$$\mathrm{SNR}_{\rm min} = 10 \log_{10} \left(\frac{\eta_i}{\sqrt{m}} \right) \ \mathrm{dB}. \qquad (3)$$

So a radar with a Pulse Repetition Frequency (PRF) of 6250 Hz that averages for 1 s would be able to detect signals 14.2 dB lower than the noise with a confidence of 99.87%. It should be noted that the *accuracy* of the measured signal is greatly reduced at low signal-to-noise ratios; see Hogan (1998)[1] for a full discussion.

There is clearly a trade off to be made between temporal resolution and sensitivity. However, an advantage of the incoherent system is that the decision on the balance between resolution and sensitivity does not need to be made at the time the data are acquired. Provided that the data are recorded at an adequately high resolution, and that noise subtraction is not performed in real time, we may defer this decision until the post-processing stage when further averaging may be performed.

2.2 Coherent processing

A coherent processing system exploits the fact that the received signal power is coherent over multiple transmitter pulses while the noise is not. The simplest technique, commonly employed by wind profilers, is to linearly average the complex received power over a defined number of pulses before calculating the various spectral moments; this removes most of the rapid noise fluctuations, while leaving the slowly-varying signal relatively intact. Unfortunately it also reduces the unambiguous velocity in proportion to the number of samples averaged, which is unacceptable for millimetre-wave radars that typically have a maximum unambiguous velocity range of only ± 5 m s^{-1}.

The alternative approach is to perform a Fourier Transform on the received complex video samples to obtain the 'Doppler spectrum', the distribution of received power as a function of velocity relative to the antenna. Although more computationally expensive, this enables the full unambiguous velocity, w_f, to be retained, while also permitting clutter suppression to be easily performed.

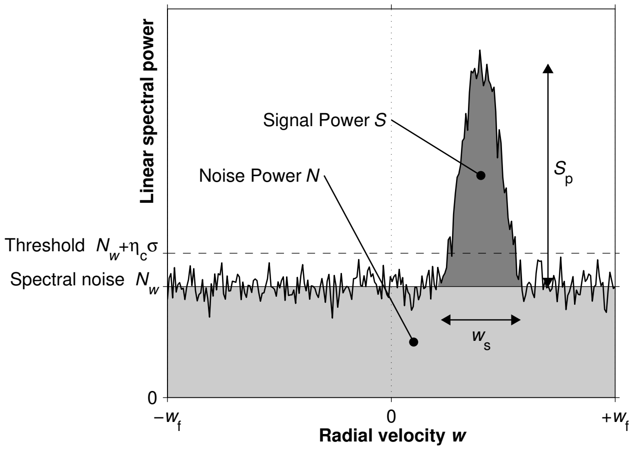

Figure 2: Idealised Doppler spectrum.

An idealised Doppler spectrum at a single gate is shown in Fig. 2. The total noise power, N, is distributed uniformly in the velocity (or frequency) domain, and is represented by the light grey area. If we define the spectral noise level to be N_w, then $N = 2w_f N_w$. The signal power, S, is the area between N_w and any significant target signal in the spectrum, as depicted by the the dark grey area. A target may have a signal too low to be detected by an incoherent processing system, but the fact that it is concentrated in a narrow range of the Doppler spectrum means that it is detectable by a coherent processing system.

The detection problem in the velocity domain is directly analogous to the detection of signals by incoherent systems in the range domain; if the spectral power in any velocity bin exceeds some threshold value then a signal is deemed to be present. We wish to minimise the fluctuations in the spectral noise power so that the threshold (η_c standard deviations above N_w) may be reduced and lower signals detected. The individual Fourier components produced by a Fourier Transform have a standard deviation equal to the mean, so we must reduce the fluctuations by incoherently averaging m_i successive spectra. This results in the standard deviation of the averaged spectral noise being reduced to

$$\sigma = \frac{N_w}{\sqrt{m_i}}. \qquad (4)$$

The minimum detectable signal then has a peak spectral signal power (indicated by S_p in Fig. 2) sufficient to just reach the detection threshold; i.e. $S_{\rm p,min} = \eta_c \sigma$. To estimate the actual signal power S that this corresponds to, we approximate the shape of the signal as a triangle in the spectrum

[1]R. J. Hogan, PhD thesis, University of Reading, Ch. 2 (available at http://www.met.rdg.ac.uk/~swrhgnrj/publications.html).

with a height S_p and a width w_S:

$$S_{\min} = \frac{1}{2} S_p w_S = \frac{\eta_c}{4\sqrt{m_i}} \frac{w_S}{w_f} N. \quad (5)$$

In logarithmic units the minimum-detectable signal-to-noise ratio is thus

$$\mathrm{SNR}_{\min} = 10 \log_{10} \left(\frac{\eta_c}{4\sqrt{m_i}} \frac{w_S}{w_f} \right) \text{ dB}. \quad (6)$$

It can be seen that signals are more easily detected if their width in the Doppler spectrum, w_S, is small.

2.3 Comparison of coherent and incoherent sensitivities

We next calculate the difference in the sensitivities of the two systems given the same target and the same number of pulses, m. Of these pulses the coherent system averages m_i spectra, each calculated from m_c pulses. Therefore from (2) and (5), the ratio of minimum detectable signals by the coherent and incoherent systems is

$$\frac{S_{\min,c}}{S_{\min,i}} = \frac{\eta_c}{\eta_i} \frac{\sqrt{m_c}}{4} \frac{w_S}{w_f} \quad (7)$$

For a fair comparison the false-alarm rate (FAR) at each range gate must also be the same. In the case of the incoherent system we assume the noise to be normally distributed, so FAR_i is the integral of a normal distribution from η_i standard deviations above the mean, to infinity:

$$\mathrm{FAR}_i = \frac{1}{2\pi} \int_{\eta_i}^{\infty} \exp\left(-\frac{t^2}{2}\right) dt. \quad (8)$$

For the coherent system, the power at only one of the m_c points in the Doppler spectrum needs to exceed the threshold for the range gate to register a false alarm. Therefore

$$\mathrm{FAR}_c \simeq \frac{m_c}{2\pi} \int_{\eta_c}^{\infty} \exp\left(-\frac{t^2}{2}\right) dt. \quad (9)$$

Due to the nature of these integrals, when we set $\mathrm{FAR}_i = \mathrm{FAR}_c$, the variables η_c, η_i and m_c are not related analytically. In the common case of $m_c = 256$ and $\eta_i = 3$, we calculate numerically that $\eta_c = 4.406$.

Considering a radar with a folding velocity of ± 5 m s^{-1}, (7) shows that the sensitivity of the two systems is the same when the full width of the signal in the spectrum, w_S, is 0.85 m s^{-1}. For narrower signals in the spectrum, the sensitivity of the coherent system is proportionately greater. If all the signal was concentrated in a single FFT bin then the coherent system would be 13 dB more sensitive.

The difference in sensitivity is only relevant for tenuous clouds that are on the margins of detection. These thin clouds tend to occupy a small part of the spectrum (certainly less than 0.85 m s^{-1}), so in most meteorological scenarios the coherent system should be significantly more sensitive than the incoherent system. Unfortunately individual insects also occupy a very narrow part of the Doppler spectrum, so the coherent system is also rather more susceptible to 'entomological clutter'. The main situation where the coherent system is less sensitive than the incoherent system is heavy rain that almost completely attenuates the radar beam. In this case the amplitude of the signal is weak but it is spread across most of the Doppler spectrum.

There are a number of extra actions performed by COHPROC which change its performance relative to an incoherent system. The most important is clutter suppression, which simply involves interpolating across the central few bins in the Doppler spectrum. In addition to removing ground clutter, this was found to remove a significant interference problem in the *Galileo* radar, which greatly compromised its sensitivity. COHPROC also allows smoothing of the spectra before extracting the signal from the noise, which further reduces the false-alarm rate.

3 The COHPROC algorithm

3.1 Overview

The algorithm itself is in the `cohproc.c` file of the COHPROC distribution. It is called once for each range gate and each ray by passing an array of m complex gate samples (i.e. I and Q) to the function `cohproc_compute`. The function returns the following computed parameters: signal power S (i.e. linear radar reflectivity factor before range correction and calibration), mean Doppler velocity $\bar{v}$ (as a real number between 0 and m_c) and spectral width σ_v (also in units of the bin size of the Doppler spectrum). The algorithm uses the FFTW library[2] to perform the Fourier Transforms. By using the FFTW data-types `fft_real` and `fft_complex`, it adopts the same precision as the FFTW library. No discernible difference in speed of execution was found between single and double precision versions.

The function optionally takes as input the mean spectrum noise level. This is for radars such as the *Galileo* where the meteorological signal can occupy more than half the unambiguous velocity range. In this situation the median of the spectrum cannot be used to estimate the mean spectrum noise level at each gate individually; rather the noise must be estimated from signal-free gates. In the case of *Galileo*, the first three gates are recorded before the pulse goes out, so are known to be free from signal; the noise estimated from these is then used in all the other gates of the ray.

[2] http://www.fftw.org

The function can also take as input the mean I and mean Q of the gate samples (i.e. the DC offset). These are only used if the `COHPROC_REMOVE_BLOCK_MEAN` precompiler macro is defined, in which case they are subtracted from all the individual I and Q values before performing any FFTs. This is a way of removing clutter, but is believed to be less satisfactory than `COHPROC_REMOVE_MEAN` (see below). The reason that the means of I and Q are passed in rather than being calculated within `cohproc_compute` is simply for computational efficiency. However, this solution is not very elegant, so if it remains unused, the `COHPROC_REMOVE_BLOCK_MEAN` option may be removed in a future version of the code.

We now describe the various stages of the algorithm.

3.2 Computation of the averaged power spectrum

The complex gate samples received by `cohproc_compute` are passed immediately to the function `compute_pxx`. The first action is to remove the DC offset, which can be done in one of three ways, depending on the settings of various precompiler macros:

1. *Fixed DC offset (`COHPROC_REMOVE_DC`).* The DC offset is taken from a file specified on the command line. This is not recommended, as the DC offset is known to drift (even for Klystron-based systems) with the result that the power spectrum can have an anomalous peak in the zero velocity bin.

2. *DC offset calculated anew each ray (`COHPROC_REMOVE_BLOCK_MEAN`).* This removes the DC component much more successfully than (1), but on the *Galileo* there have still been puzzling problems with noisy rays, which imply that the DC offset can sometimes change significantly within a ray (i.e. on a timescale of less than a second).

3. *DC offset calculated anew before each FFT (`COHPROC_REMOVE_MEAN`).* This forces the DC component of each spectrum to be zero, except for the slight smearing effect of the window function used. The resulting notch in the spectrum at the zero velocity bin can be removed easily with the interpolation option, described in section 3.5.

As discussed in section 2.2, the m input samples are split into m_{i} groups of m_{c}, and each group is processed in turn. If the `COHPROC_WINDOW` precompiler macro is defined, then the m_{c} samples are convolved with an FFT window to avoid spurious sidelobe effects in the spectrum. Currently a Blackman window is used (see section 4.4). An FFT is performed on each group of m_{c} samples, and the resulting m_{i} spectra are averaged. This is more computationally efficient than performing a single m-point FFT and then smoothing it in the frequency domain.

The `compute_pxx` function also calculates the location of the highest spectral power in the domain.

3.3 Optional recording of raw spectra

An averaged but otherwise untreated Doppler spectrum is returned from `compute_pxx`. COHPROC allows for periodic recording of profiles of these raw spectra, and at this point it can copy the raw spectrum into a separate buffer, which is later written to a separate file using the functions in `spec.c`.

3.4 Estimation of the noise level

The mean noise level of the spectrum may now be estimated simply from the median of the spectrum. This is valid provided that the signal occupies less than half the spectrum. For radars with a low folding velocity, the noise level should be calculated in this way only for gates known to be signal-free; the values calculated in these gates can then be used in all the others.

3.5 Elimination of clutter

Clutter suppression is performed in the simplest possible way by the `interpolate_over_clutter` function, which interpolates over a specified number of points to each side of the zero-velocity bin of the Doppler spectrum, thereby removing any spike or notch there. To eliminate mirror-image peaks, the same number of points are interpolated over at the folding velocity, 180° away in the spectrum. A previous, more complicated, clutter suppression algorithm was found to remove far too much good signal.

3.6 Smoothing of the spectrum

The spectra may next be smoothed in the velocity domain using a simple boxcar average, which further reduces the false-alarm rate.

3.7 Signal identification

If either smoothing or clutter suppression have been enacted, then the location of the highest power in the spectrum (originally computed in `compute_pxx`) is recalculated. The algorithm then moves down in both directions to either side of the peak in the spectrum until it reaches the threshold level. The threshold is currently set at an arbitrary 1.45 times the level corresponding to a 99% confidence limit (see section 4.5). It records the positions of the extremes of the signal in the variables `left` and `right`. In

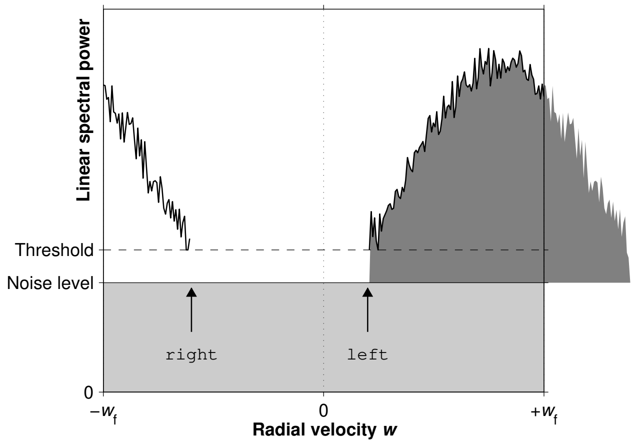

Figure 3: Schematic illustrating the unfolding of a Doppler spectrum that does not fill the velocity domain. The thick line shows the original processed spectrum, and the dark shaded area shows the unfolded spectrum that is used to calculate radar reflectivity and the Doppler parameters.

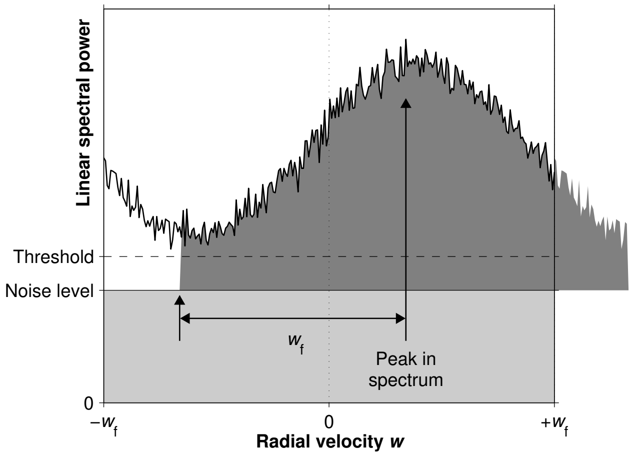

Figure 4: Schematic depicting the unfolding of a Doppler spectrum that occupies the entire velocity domain. The thick line shows the original processed spectrum, and the dark shaded area shows the unfolded spectrum that is used to calculate the radar parameters. The spectrum is folded at a point w_f (i.e. 180°) away from the peak in the spectrum.

each bin between these limits the mean spectrum noise level is subtracted. The code correctly identifies folded spectra that do not fill the entire velocity domain; in these cases `left` $>$ `right`. If there is signal in only a single bin, or if no signal is detected at all, then `left` $=$ `right`.

3.8 Computation of radar parameters

Lastly the function `comp_params` computes the radar parameters from the processed spectrum, `left`, `right` and the location of the highest value in the spectrum.

It first tests whether `left` $=$ `right`, in which case a meteorological signal is deemed not to have been detected, and a value of -1 is returned for S. Hence COHPROC rejects signals concentrated entirely in a single bin, which are usually due random fluctuations in the noise or to insects. In reality, the use of the Blackman window and subsequent smoothing in the spectral domain means that single peaks at this stage in the processing are very uncommon.

If the signal does not fold (i.e. `left` $<$ `right`), then S is simply the sum of the signal in each bin, s, between `left` and `right`:

$$S = \sum_{k=\texttt{left}}^{\texttt{right}} s_k \qquad (10)$$

Mean Doppler velocity and spectral width are similarly calculated as the normalised first and second moments of the distribution, respectively:

$$\bar{v} = \frac{1}{S} \sum_{k=\texttt{left}}^{\texttt{right}} k s_k \qquad (11)$$

$$\sigma_v = \left(\frac{1}{S} \sum_{k=\texttt{left}}^{\texttt{right}} k^2 s_k - \bar{v}^2 \right)^{\frac{1}{2}} \qquad (12)$$

Note therefore that the spectral width is simply the standard deviation of the distribution. For a perfect Gaussian, this is equal to $(2\ln 2)^{\frac{1}{2}}$ times the half-width of the distribution at the level of half the peak power.

If the signal folds then it needs to be first unfolded in order that the Doppler parameters are unbiased. The first case considered is a signal that folds but does not fill the entire spectrum. This is indicated by `left` $>$ `right`, and is dealt with simply by rejoining the separated parts of the signal as depicted in Fig. 3. The other case considered is a signal that fills the entire spectrum, indicated by `left` $= 0$ and `right` $= m_c - 1$. In this case the spectrum is folded at the point w_f away (i.e. 180° away) from the maximum in the spectrum, as shown in Fig. 4. If the unfolding of the spectrum results in $\bar{v}$ lying outside the range 0 to m_c then it is refolded back in.

4 Scope for improvement of the algorithm

4.1 Occasional loss of sensitivity relative to incoherent processing

As discussed in section 2.3, coherent processing can be less sensitive than incoherent when the spectral width of the target is large but the signal power is small. This problem has been observed at 94-GHz in strongly attenuating rain. A solution would be to also perform incoherent calculations of S, $\bar{v}$ and σ_v, and to use them in cases where the coher-

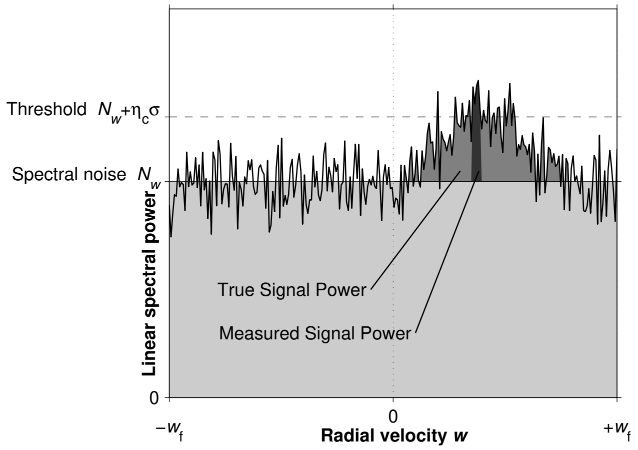

Figure 5: Schematic illustrating why COHPROC tends to underestimate both signal power and spectral width when the signal-to-noise is low but the target is wide.

ent algorithm was unable to detect a signal. This would unfortunately result in 'step changes' in the various parameters because of the different biases of the two systems at low signal-to-noise ratio. Currently the PCI-based *Galileo* data acquisition system records the incoherent signal power, S, in a separate file from the parameters produced by the COHPROC algorithm. Given the low additional computational cost, it would be useful if incoherent $\bar{v}$ and σ_v could also be recorded (see Srivastava et. al 1979[3] and Chapman and Browning 2001[4]). A further important improvement to the incoherent recording would be to remove the DC offset before calculating any parameters, since the interference in the 94-GHz radar exhibits itself as a DC offset that varies in both time and range, and significantly reduces the absolute sensitivity of the system.

4.2 Inaccurate signal and width estimation at low signal-to-noise ratios

Even if a low signal with a relatively large spectral width is detected by the coherent algorithm, it can underestimate both S and σ_v because it does not consider the shape of the spectrum below the threshold level. This is demonstrated by the schematic in Fig. 5, in which the measured signal power is much narrower and has a much smaller area than the true signal power. Again, the only solution would seem to be to simultaneously calculate the same radar parameters incoherently, and then intelligently select which system is more likely to be accurate in each case (or even use a weighted average of the values calculated from the two systems).

[3]Time-domain computation of mean and variance of Doppler spectra. *J. Appl. Meteor.,* **13,** 472–480.

[4]Measurements of dissipation in frontal zones. *Quart. J. Roy. Meteorol. Soc.,* **127,** 1939–1959.

4.3 Clutter suppression

The current system of interpolating over a fixed number of bins in the centre Doppler spectrum is effective for removing several types of spurious signal, but at close range the ground clutter tends to spill into more gates and so is not fully removed. A simple improvement would be to interpolate over more bins in the lowest few gates in order to fully eliminate the clutter. The actual number at each gate would be different for each radar and would have to be decided by looking at some sample data. An earlier version of COHPROC attempted to reject clutter by working down the two sides of the central peak until it reached the noise floor, and removing everything as it went. In practice this removed a large amount of good signal.

4.4 Use of the Blackman window

Currently a Blackman FFT window is used, which has a low sidelobe level of -57 dB, but a relatively high leakage of sharp spectral features into neighbouring bins. Since the dynamic range of the spectra is usually much less than 57 dB, it may be worth considering the adoption of a window with less leakage into neighbouring bins, at the expense of a higher sidelobe level. This way the number of central bins that one needs to interpolate over to remove clutter could be reduced.

4.5 Threshold levels and averaging in the spectral domain

Currently COHPROC uses a detection threshold that is 1.45 times the 99% confidence level. This value was chosen empirically at a time when COHPROC was not even eliminating the noise correctly, and it does not change in response to the amount of boxcar smoothing performed on the spectrum, despite the fact that smoothing effectively changes the confidence level by reducing the noise fluctuations. In practice, the amount of smoothing performed is chosen purely to keep false alarms to a minimum, while not compromising the sensitivity. Rather than performing smoothing in the spectral domain it would be more efficient to average a larger number of spectra, each calculated from fewer pulses. It would also be useful if the confidence level could be set on the command line.