## Initial Evaluation of Oversampled PFB inversion by FFT for Pulsar Timing Applications

Ian Morrison, August 2016

**Summary**: This report describes a method for inverting an oversampled polyphase filterbank (PFB) using FFT and inverse-FFT processes, as a means of obtaining higher time-resolution signals from the combination of multiple neighbouring and overlapping fine channels from a PFB. This approach is of particular relevance to narrowband beamforming for pulsar timing, since it has the potential to eliminate the need for a synthesis filterbank after beamforming for generating the coarse channels used for pulsar timing purposes. Instead, the fine channels can be supplied to the pulsar timing engine, which can itself reconstitute coarse channels of any desired time resolution. Furthermore, the process can be combined with the coherent de-dispersion process, meaning there is no additional computational burden on the pulsar timing engine (in fact there is a reduction from replacing a single large FFT with several smaller ones). If this technique proves feasible, then it could also prove to be a more computationally efficient and flexible method than true time-delay beam formation.

This report describes Matlab models developed to evaluate the technique’s suitability for pulsar timing purposes, and presents simulation results for an illustrative 8-channel PFB that confirm the basic utility of the method, subject to certain constraints on the PFB design. It was also observed that, to simplify the signal reconstruction process, a specific alignment should be employed between the PFB and the forward FFTs of each fine channel. The origin for such a favoured alignment is explained, together with its implications for the CBF/PST system design.

### The Problem

The correlator/beamformer (CBF) for SKA.Low will implement narrowband beamforming for the pulsar timing (PST) beams. This requires that LFAA coarse channels be further fine-channelised to a resolution of the order of a few kHz. Beamforming delays can then be achieved by applying a phase gradient across the fine channels, implemented by multiplying each fine channel with the appropriate complex unit phasor. These fine channels have insufficient time resolution for pulsar timing purposes, so normally a synthesis filterbank would be needed after beamforming to reconstitute wider coarse channels to pass to the PST engine.

The first task performed by the PST engine is coherent de-dispersion using a convolving filterbank in the frequency domain, requiring that input coarse channels are first FFT’d. It was recognised that the CBF synthesis filterbank might be completely eliminated if the spectra of fine channels produced by the CBF could be combined within the convolving filterbank employed by the PST. If so, there is an additional benefit in that obtaining the required wide bandwidth spectrum can be achieved with a number of shorter FFTs of each fine channel rather than a single large FFT of the full coarse channel, providing a theoretical reduction in computational burden. However, on some multi-processor architectures, such as the Graphics Processing Units that will be employed by the PST, larger FFTs may execute more efficiently than many smaller FFTs. Therefore, some benchmarks should be performed to assess the computational performance of PFB inversion during convolving filterbank formation in comparison with convolving filterbank formation on coarse channels.

### The PFB Inversion Method

The use of oversampling in the fine channeliser offers the possibility of combining fine channels using FFTs rather than a true synthesis filterbank. With a suitably designed OS-PFB, all the spectral leakage between adjacent fine channels will be contained within transition bands that correspond to the overlapping portions of adjacent oversampled fine channels. After an M-point FFT of a fine channel, samples at the band edges corresponding to the transition bands can be discarded, leaving a spectral segment of length L=M/FOS (where FOS is the oversampling factor) representing the pass-band of the fine channel, with width equal to the fine channel spacing. Processing each fine channel in this manner provides a set of spectra that represent contiguous sub-band segments of the coarse channel spectrum. Concatenating N such segments should produce the equivalent spectrum to what would be obtained from an NxL-point FFT of the N times wider bandwidth input signal.

It is important to note that this equivalence is only strictly true when: (i) the fine channel response is completely flat across its pass-band, (ii) all spectral leakage has been eliminated (i.e. contained entirely within the overlapped portions of the oversampled fine channels, which are discarded), and (iii) there are no quantisation or other imperfections in the implementation. Of course, none of these conditions is ever true in practice, so there will always be some degree of error in the inversion process. If one were to perform an IFFT on the concatenated spectrum and compare this reconstructed time series to the original input into the PFB, there will always be some degree of mismatch. The goal is to select parameters for the PFB design and inversion process that result in an acceptable level of error.

It is also important to note that verification of behaviour for a single PFB followed by inversion addresses only part of the problem. With beamforming there are multiple antenna stations/dishes, each with their own PFB, and each PFB’s outputs will have a different phase-gradient applied before being summed on a fine-channel-by-fine-channel basis. The inversion is applied to a group of summed fine channels. It may be expected that the FFT inversion process is linear and therefore should behave equally well for the single or multiple-summed PFB cases. However, it is recommended that this be validated by simulation of a multiple station/dish scenario to confirm that the application of different phase gradients prior to summing does not produce any surprises.

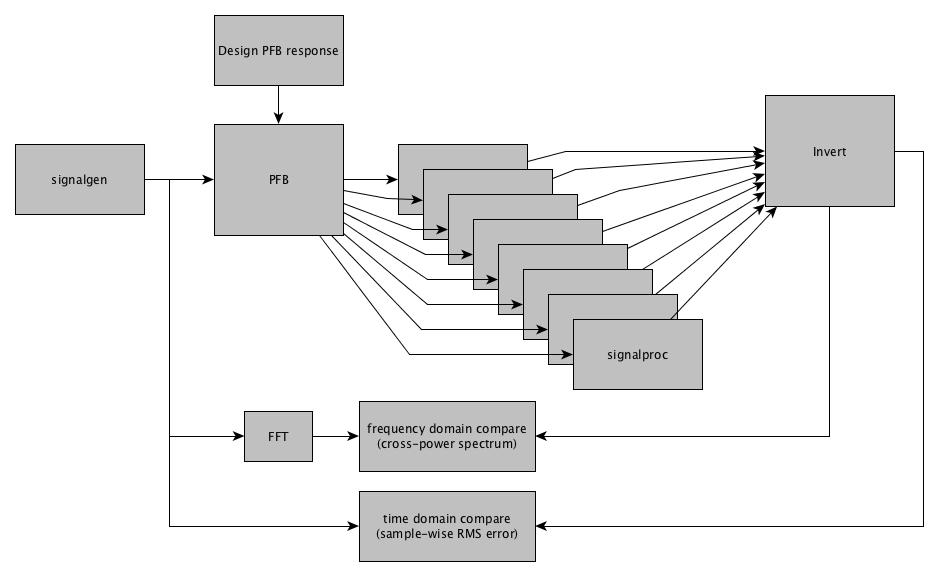
### Inversion Performance

Establishing what is an acceptable level of inversion error for pulsar timing purposes is not straightforward. The characteristics of the different artefacts introduced by imperfect inversion are not well understood, and it is difficult to establish a single performance metric to quantify the overall performance degradation. Relative measures can be obtained in various ways, such as measuring the sample-by-sample RMS error between the reconstructed and original waveforms for a range of different parameter choices. However, it is unclear how to relate such measures to the resulting impact on pulsar time-of-arrival (ToA) estimation accuracy – which is the end game. Ultimately the most comprehensive test method would be to process a simulated (or measured) pulsar signal and measure the actual effect on ToA by comparing the results with and without the PFB/inversion stages present. To accomplish this requires folding a long sequence of pulses, which means the simulation system must be able to operate in a continuous manner. This has not been implemented at this point in time – the system currently processes one single block of data at a time. Continuous operation is a recommended extension.

### Matlab Model – single PFB

The architecture of the Matlab simulation model is depicted below. It is implemented with the following functions:

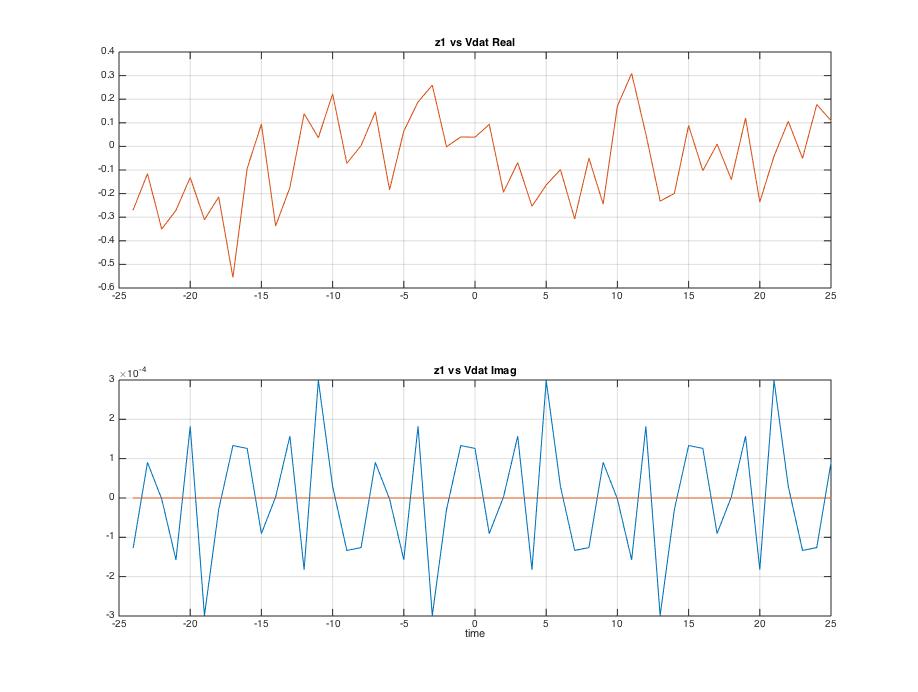
1. main\_script.m – sets the simulation parameters and calls the functions below.
2. signalgen.m – produces a test signal, either an impulse or simulated pulsar signal.
3. Design\_N\_channel\_OS\_8\_7\_PFB.m / Design\_N\_channel\_OS\_4\_3\_PFB.m – creates the impulse response for the PFB (only FOS = 8/7 or 4/3 supported at this stage).
4. PFBchannelizer.m – implements the PFB, with a configurable impulse response.
5. signalproc\_phase\_shift.m – performs the forward FFT of a fine channel, with an optional phase shift beforehand. Also optionally performs PFB pass-band ripple equalisation.
6. invert\_N\_channels.m – collects the fine channels, concatenates and reconstructs the signal. Also performs comparison of input and output, both time-domain (RMS error) and frequency-domain (cross-power spectrum).



### Example Inversion Simulations

To illustrate the inversion process, here we show results for an 8-channel PFB with FOS = 8/7 and with a simulated noisy pulsar signal as input. 32-bit floating-point data/variables are used throughout. The input is real, so the imaginary component of the output should be zero in the ideal case. The forward FFT length used in all cases is 1024. The orange plots are the input waveform and blue is the reconstructed output.

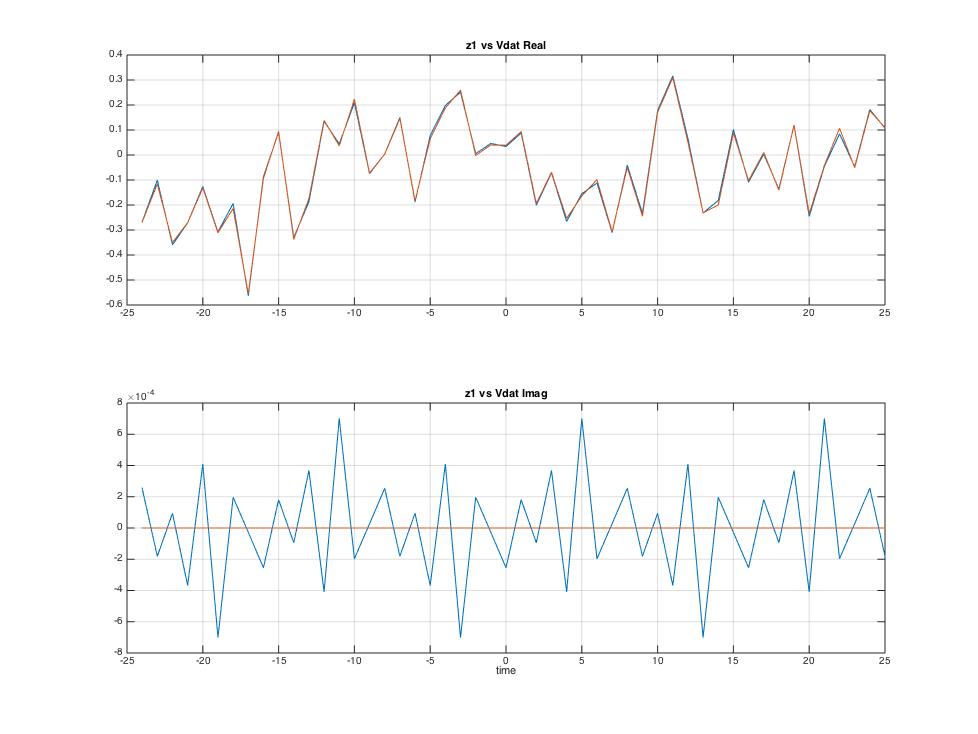
**PFB impulse response length = 256 (32 taps per channel)**



To the naked eye, there is no visible difference between the real components of the reconstructed and original waveforms, i.e. they are fully overlaid. There is some pollution of the imaginary component, not exceeding a peak amplitude of 3x10-4. The average RMS error per sample, measured over an interval of 200 samples, was found to be 3.5x10-05 (real) and 1.5x10-04 (imag).

These results suggest that for this (large) number of PFB taps, the pass-band is very flat and spectral leakage has been effectively eliminated.

**PFB impulse response length = 64 (8 taps per channel)**

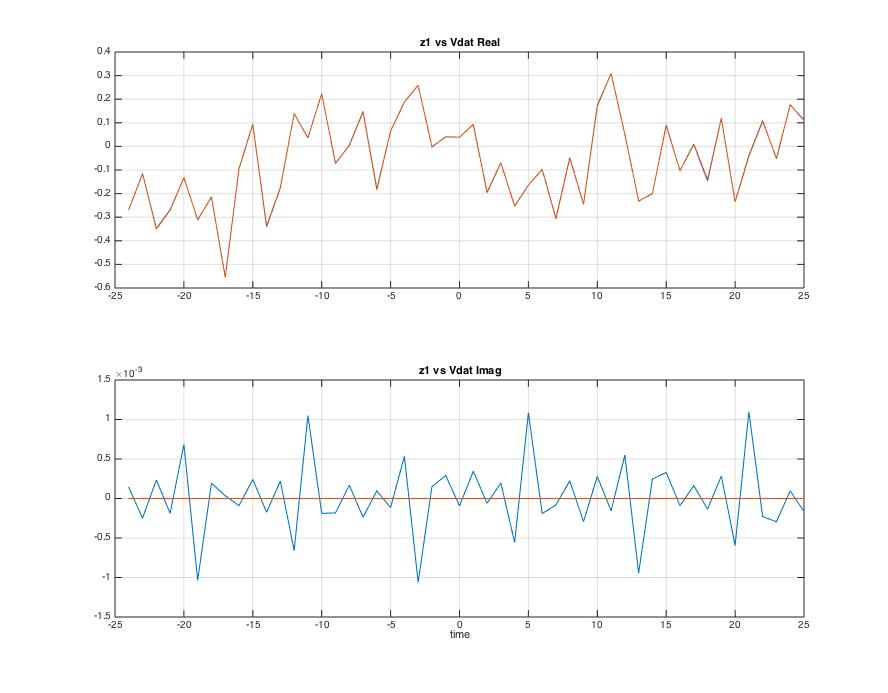


With much fewer taps, we now start to see a visible difference between the real components of the reconstructed and original waveforms. The pollution of the imaginary component has also increased to a peak amplitude of ~7x10-4. The average RMS error per sample, measured over an interval of 200 samples, was found to be 8.3x10-03 (real) and 3.4x10-04 (imag).

These results suggest that for this (small) number of PFB taps, the pass-band ripple is significant and/or there is some residual spectral leakage.

It is possible to isolate the effects of pass-band ripple and spectral leakage by applying an equalisation process to the fine channel spectra prior to combining them. This is accomplished by dividing each complex element of the spectrum by the magnitude of the PFB transfer function at the corresponding location in the frequency response. The result for 8 taps-per-channel with equalisation is shown below.

**PFB impulse response length = 64 (8 taps per channel) + pass-band equalisation**



With equalisation there is again virtually no visible difference between the real components of the reconstructed and original waveforms. The pollution of the imaginary component, however, has increased slightly to a peak amplitude of ~1x10-3. The average RMS error per sample, measured over an interval of 200 samples, was found to be 1.9x10-03 (real) and 4.6x10-04 (imag).

These results suggest that most of the degradation with 8 taps is due to pass-band ripple rather than spectral leakage. Equalisation has been very effective in this case, giving results that are visually comparable to the 32-taps-per-channel case, although the RMS errors are somewhat higher. Implementing the equalisation as part of the inversion process is trivial, but leads to a major reduction in the computational complexity of the PFB; from 32 to 8 taps-per-channel.

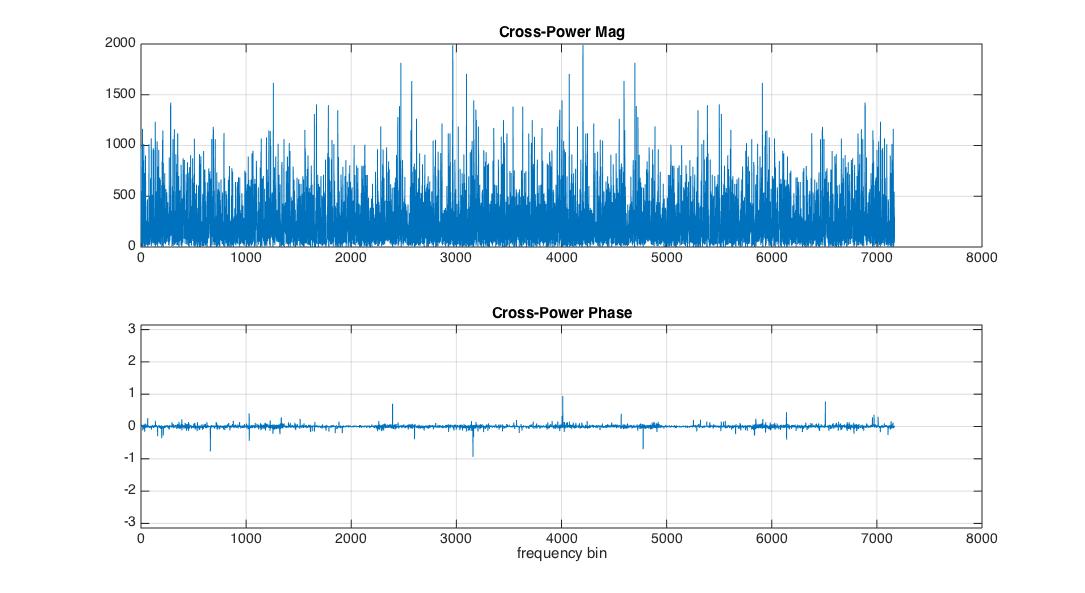
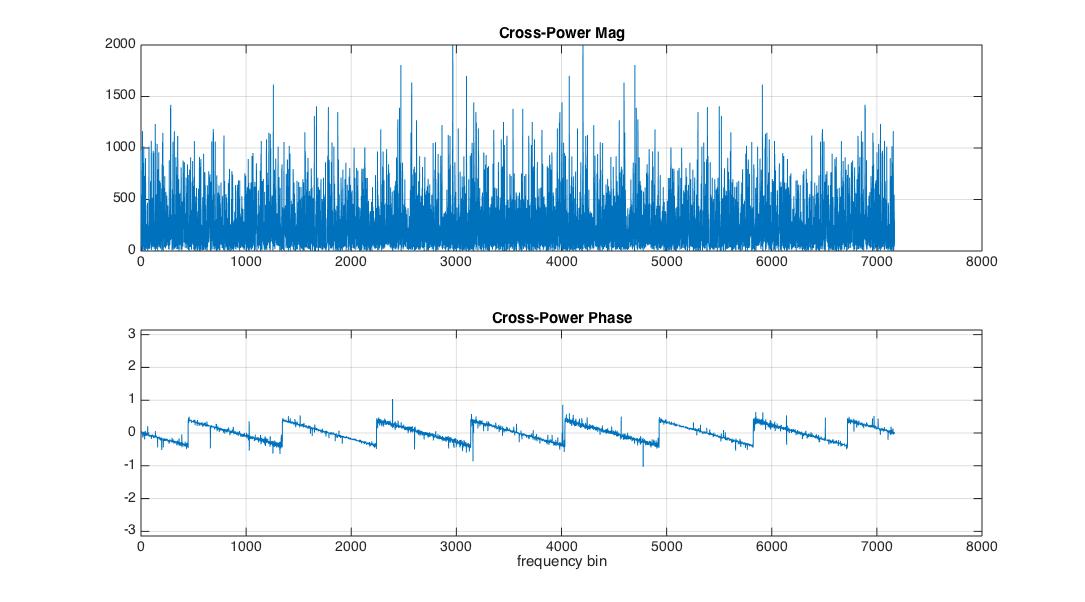
### PFB/FFT Alignment

In the course of developing the Matlab model it was discovered that there exists a favourable relationship in the relative temporal alignment between the PFB and fine channel forward FFTs. The favourable alignments in terms of the start time index for the FFT are modulo-8 in fine channel sample index when FOS = 8/7, and modulo-4 in fine channel sample index when FOS = 4/3. The absolute value for a preferred alignment depends on the length of the PFB impulse response, which is understandable because the PFB introduces a delay of half the impulse response length, so changing the length alters the relative delay between the PFB and FFT.

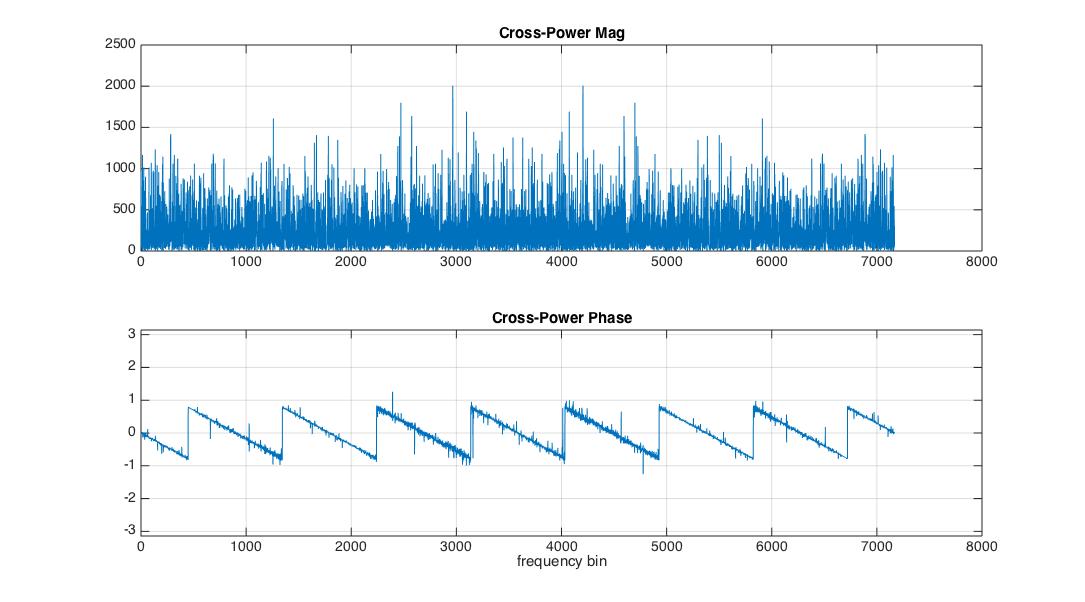
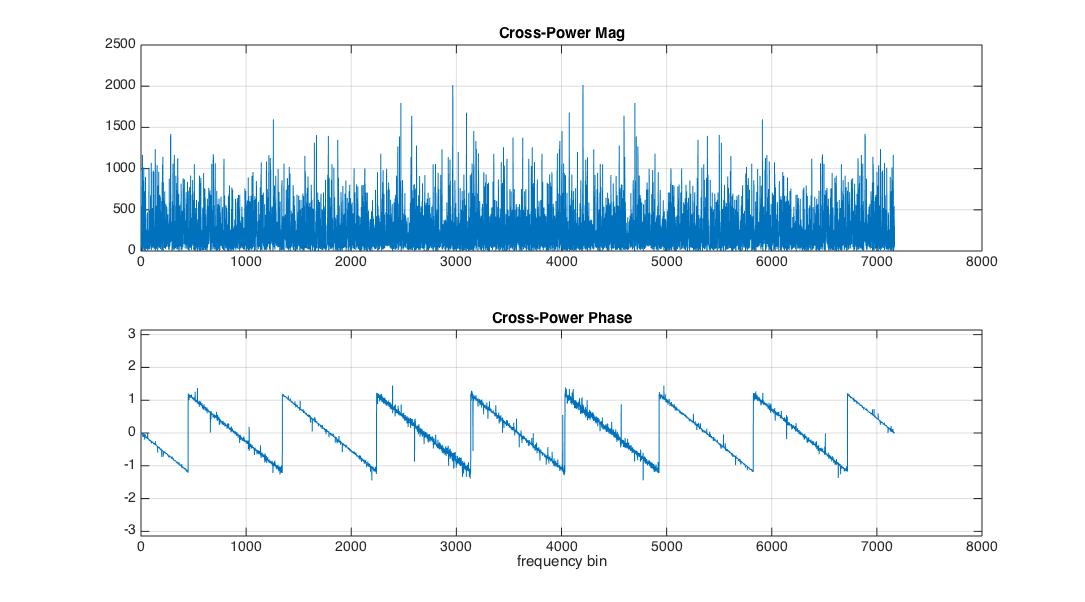
The favourable alignment is not a fundamental requirement for correct signal reconstruction, but it does introduce a very attractive simplification – which is that no phase rotations need to be applied to individual fine channel spectrum segments before they are concatenated. For other alignments there is a need to apply an additional phase rotation to each segment prior to concatenation, of an amount that depends on the offset from the favoured alignment. At the favoured alignment, all the phase rotations become multiples of 2π, hence effectively disappear.

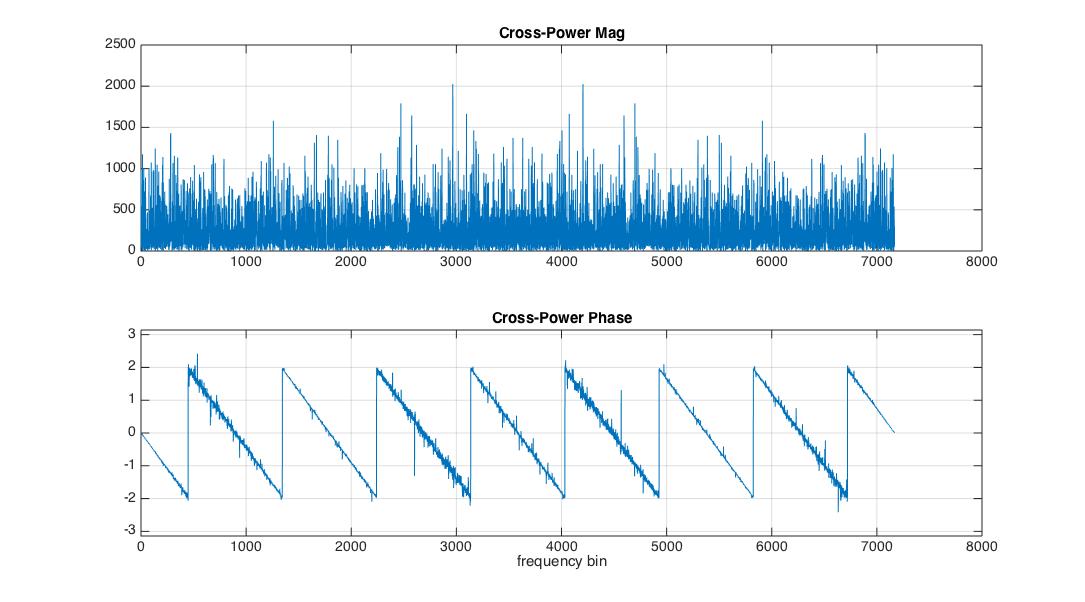
This behaviour can be understood by recognising that there will be a difference in the FFT output in the case when (i) the centre of the PFB impulse response falls exactly on a fine channel sample instance, and (ii) when it does not. With an N-channel PFB, there is a spacing of N between taps that combine to form each of the N inputs to the PFB’s internal FFT block. If the delay between the PFB and fine channel FFT boundary shifts by one impulse response sample interval (which will happen if the impulse response length is incremented by two), then this corresponds to a fractional shift on the fine channel of 1/Nth of a sample interval. The resulting FFT output will have essentially the same magnitude response but a phase response with a gradient of 2π/N across its spectrum. This is illustrated by the example below for an 8-channel PFB with FOS = 8/7. Here the cross-power spectrum between the input and output has been plotted for nine different alignment cases (corresponding to impulse response length increments of 0, 2, 4, 6, 8, 10, 12, 14 and16). Each plot shows the concatenation of the 8 fine channels, prior to the IFFT to reconstruct the time series.

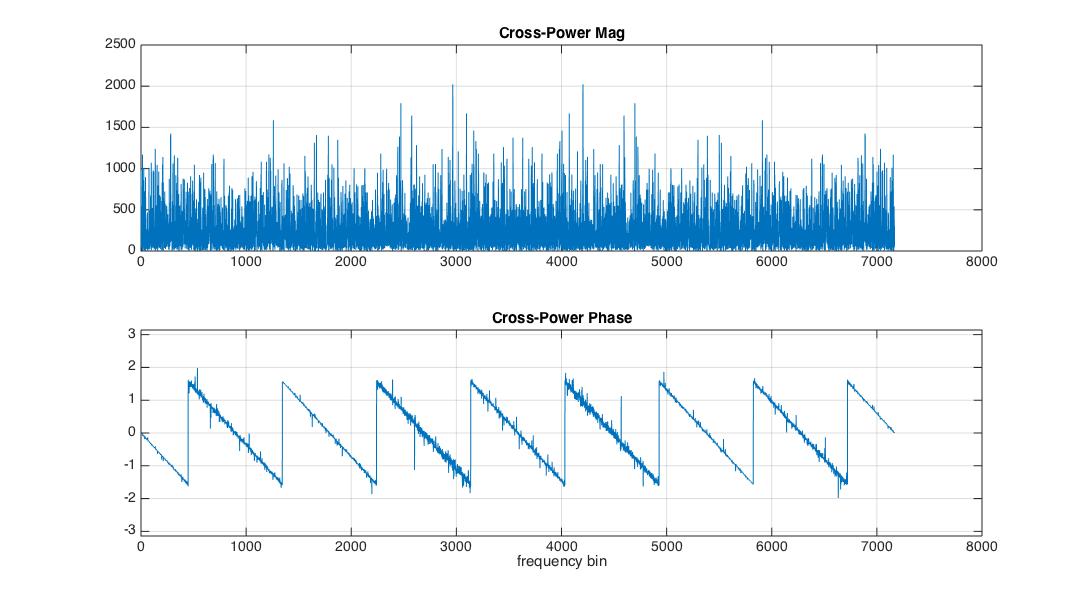
**Impulse Response Length Increment = 0 Impulse Response Length Increment = 2**



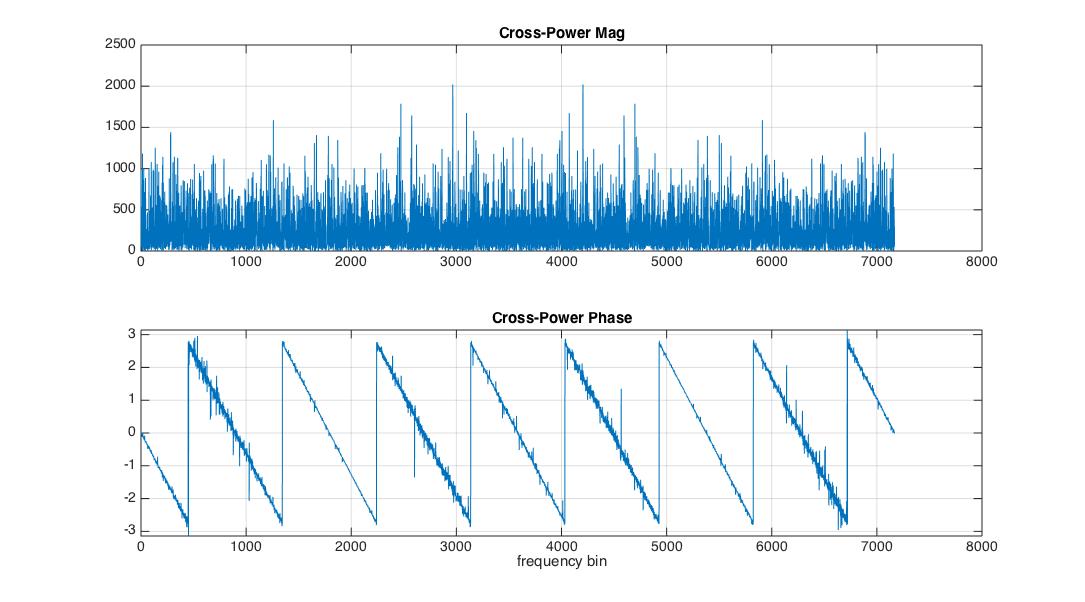
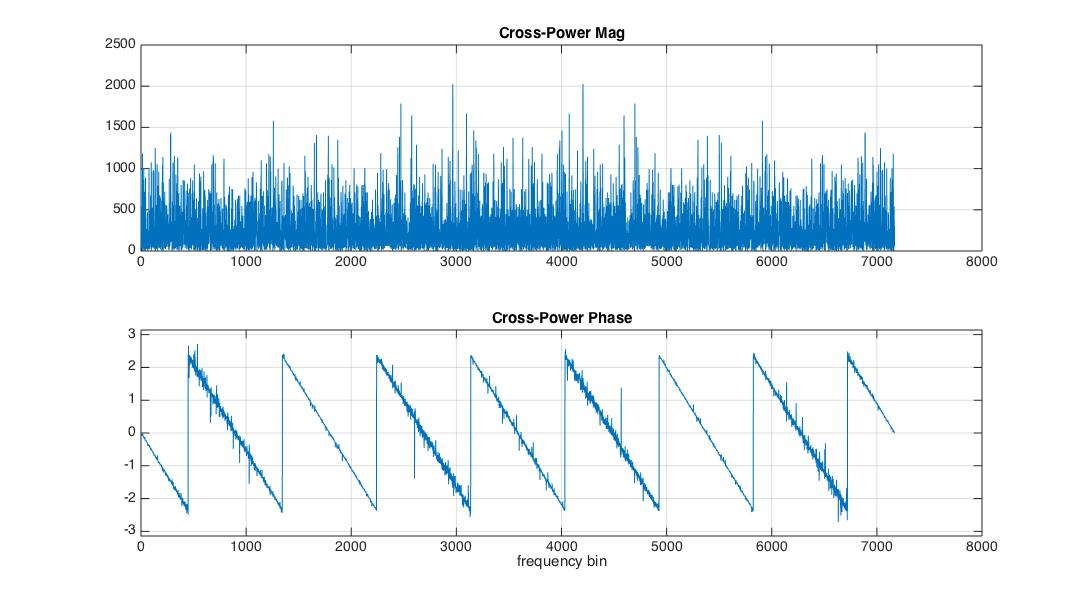
**Impulse Response Length Increment = 4 Impulse Response Length Increment = 6**



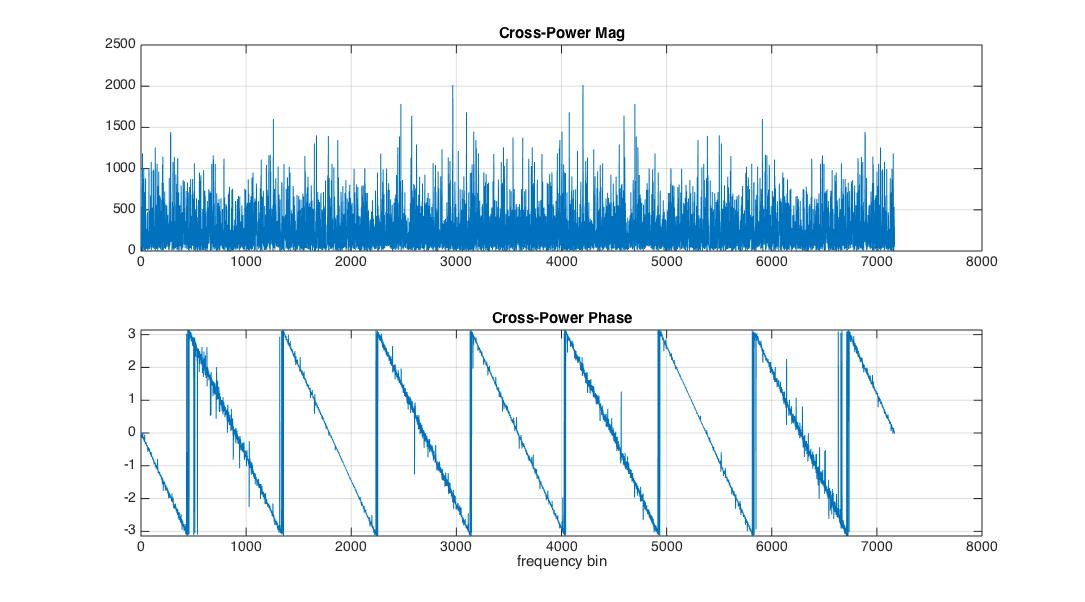
**Impulse Response Length Increment = 8 Impulse Response Length Increment = 10**



**Impulse Response Length Increment = 12 Impulse Response Length Increment = 14**

****

**Impulse Response Length Increment = 16**



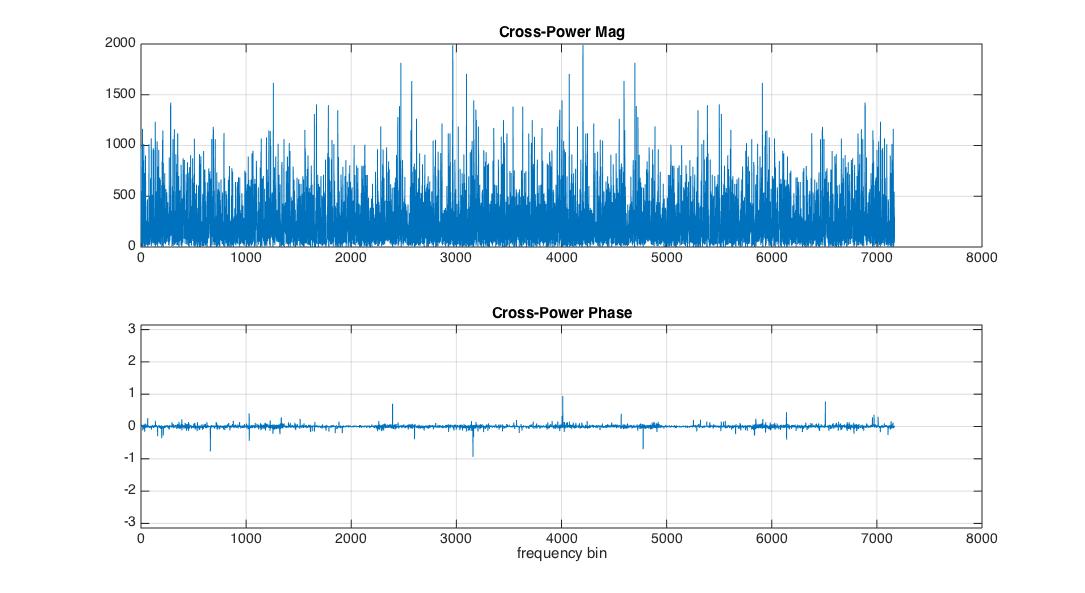
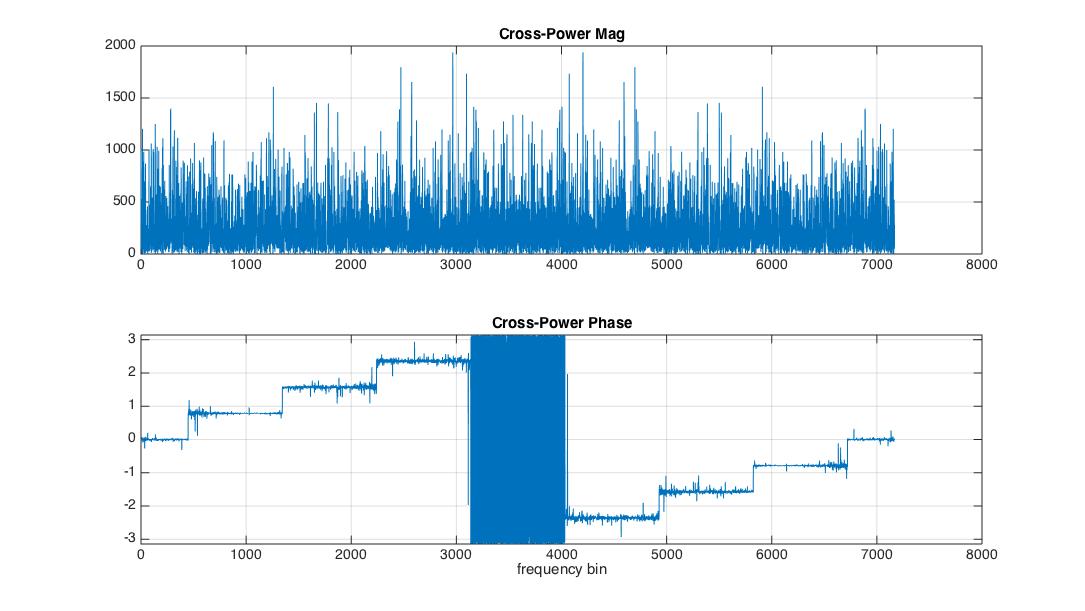
The 16-increment case corresponds to the impulse centre shifting by exactly one fine channel sample, hence we would expect to see 2π phase gradient across each fine channel – which is precisely what is seen.

For other cases the effective fine channel shift is less. For the 2-increment case the fine channel phase gradient is 2π/8. For the 4-increment case it is 4π/8, and so on. In general, for 2M increments (M impulse response sample shifts), the fine channel phase gradient is M.2π/8. When the fine channel spectral segments are concatenated, phase continuity is disrupted unless each segment is first phase rotated by the appropriate multiple of 2π/8 (in this example). For the Mth alignment shift, the required phase rotation is K.M.2π/8 where K is the channel index. When M gets to 8, the rotation is K.2π, which is equivalent to zero on every fine channel. Hence every 8th alignment (a total increment of 16 in impulse response length) is favourable because it eliminates the need for any phase rotations prior to concatenating each fine channel segment.

Now consider the case where the length of the PFB impulse response is kept fixed and we move the start index of the fine channel forward FFTs. We can only increment by whole numbers of fine samples. We want the impulse response centre to fall exactly on a fine channel sample instance. For the alignment where this happens, there is no fractional delay and corresponding phase gradient. Consider a shift of the FFT start by one whole fine sample. For 8 channels and 8/7 oversampling, each fine channel sample interval corresponds to 7 sample intervals of the input signal, not 8. So a single-sample shift on the fine channel introduces one impulse response sample of mis-alignment between the impulse response centre and the fine channel sample instances. Therefore it will take 8 such fine channel shifts before the impulse response sample mis-alignment reaches 8, again placing the impulse response centre at the fine channel sample instance. So, in a similar way to when the impulse response length is varied, there is a modulo-8 relationship in the FFT start index between favourable alignments. For 4/3 oversampling the favourable alignments will have a modulo-4 relationship. In between the favourable alignment cases, the individual fine channel spectrum segments will need to be phase rotated before concatenation.

Another perspective on this behaviour can be seen by analysing the cross-power spectrum after applying the appropriate delays between input and output. As mentioned, a single sample time shift on a fine channel corresponds to exactly 7 coarse channel sample shifts (for the FOS = 8/7 case). Below are the cross-power spectra for the correctly aligned case and the case where there is one sample shift to the fine channel FFTs, and the comparison of the output is made with the input coarse channel signal that has had 7 sample shifts applied.

**FFT start shift = 0, coarse input shift = 0 FFT start shift = 1, coarse input shift = 7**



Referring to the right side plot, it is seen that the phase has remained flat within each fine channel, as one would expect given the input and output delay shifts have been matched. However, it is also seen that each fine channel now has a 2π/8 phase rotation, due to the impulse response mis-alignment. For M fine channel shifts, the phase rotation is M.K.2π/8, where K is the channel index. When M gets to 8, the rotation is K.2π, which is equivalent to zero on every fine channel. Hence every 8th alignment (8 increments of the FFT start index)) is favourable because it eliminates the need for any phase rotations prior to concatenating each fine channel segment.

In a practical implementation of this method of OS-PFB inversion, it is recommended that the existence of favourable PFB/FFT alignments be exploited to eliminate the need for performing phase rotations of individual fine channel spectrum segments. This means the packing of fine channel samples into the packets delivered by CBF to PST should be deliberately chosen such that the PST knows where to start its FFT boundaries. If the PFB design has the flexibility to vary the length of its impulse response, this should to be factored into the packetising process such that the same FFT alignment applies at the PST regardless of what length impulse response has been chosen by CBF.

The boundaries for favoured alignments on the PFB outputs will be fixed by the PFB hardware design. When samples for each fine channel are buffered for packetisation on the CBF-to-PST interface, they can be placed in buffer memory such that the first sample of each packet represents a favourable alignment case.

### Other Issues/Concerns

1. With beamforming, the PFBs for all stations/dishes will need to operate in phase so that a common alignment relationship exists between the fine channel forward FFTs (which are applied post-summation) and each individual PFB. Otherwise there will be different phase gradients present in each of the constituent signals being summed, and then there can be no single set of phase rotations to each fine channel prior to concatenation to provide phase continuity across the reconstituted coarse channel.
2. It may be tricky to reconstruct a coarse channel in the PST that is wider than the LFAA coarse channel width. There are two PFB stages involved, which will complicate the FFT alignment problem.
3. The favourable fine channel FFT alignments derive from the numerator of the OS factor. (This has been confirmed for 8/7 and 4/3, but should also be confirmed for other OS factors.) The choice of a smaller OS numerator like 4 or 8 will provide greater flexibility in the choice of FFT length and also simplify the design of the output sample buffering for packetisation. Both the FFT length and packet length should be a multiple of the favourable alignment spacing, so that the alignment is maintained across packets and FFTs.
4. For continuous inversion, some form of zero padding or overlap-save method must be used to accommodate the PFB filter memory when processing samples in blocks. This is another reason to favour an OS factor that gives a closer spacing between preferred alignments, since it will provide finer-grained choice of the overlap length.

### Proposed Model Extensions

1. **Generic number of PFB Channels**
2. **Continuous Inversion** – implement the overlap-save method to deal with the convolution overlap issue arising due to PFB filter memory.
3. **Generic OS Factor** – correct operation with any OS factor, and verification of the corresponding PFB/FFT alignment constraints.
4. **Beamformer Demonstration** – multiple PFBs, phase-gradients and summation prior to inversion.
5. **Pulsar Timing Performance** – folding of multiple pulses, measurement of ToAs, and comparison against the case with no PFB/inversion.