ON RECONSTRUCTING FILTERED TIME SERIES FROM A SEQUENCE OF OVERLAPPED, WINDOWED, AND FOLDED FAST FOURIER TRANSFORMS

fred harris

Electrical & Computer
Engineering Department
San Diego State University
San Diego, CA 92182

CODE 733, BLDG 607
Signal Processing & Display
Naval Ocean Systems Center
San Diego, CA 92152

ABSTRACT

A common signal processing task, related to fast convolution, is the reconstruction of a time series from a set of spectral data obtained from the output of a DFT and subjected to a specified spectral modification. With post processing, similar to echo cancelling equalization, the time series can be reconstructed from alternate (ie. the even indexed) filters of a windowed DFT. This reconstruction is possible because the high degree of correlation in the adjacent overlapped filters of a windowed DFT assures us that the data associated with the missing DFT bins is present in (and can be extracted from) the adjacent DFT filters.

INTRODUCTION

Narrow band time series analysis can be performed with a sequence of overlapped and windowed DFTs. This form of analysis is applied to the design of transmultiplexers. In this application, the DFT simultaneously demultiplexes and resamples a set of FDM channels. The center frequencies of the filter bank are the DFT bin frequencies f_s/N , where N is the size of the DFT. The bandwidth of the individual filters is set by the window which, in this application, is the impulse response of the prototype low pass FIR filter. The resampling ratio of the equivalent FIR filter bank, which is limited to the ratio of the input to output bandwidth, is controlled by the shift interval (or overlap) between successive transforms. The unwrapped DFT filter bank has a Dirichlet kernel (ie. $\sin x/x$ like) spectral response. The bandwidth of these filters exactly match the channel spacing of the transform, f_s/N . This response, shown in figure 1, is the narrowest possible bandwidth for the N points of input data but is characterized by unacceptably high sidelobes. The high sidelobes represent channel crosstalk or aliasing terms which fold back into the mainlobe response under the resampling operation.

The windowing operation applied to the data controls the analysis sidelobe levels and has the incidental effect of broadening the mainlobe response of the filter bank. The mainlobe bandwidth is increased from f_s/N to $K(A) f_s/N$. Here $K(A)$, the broadening factor, is approximately $A/20$ where A is the filter peak to sidelobe level ratio in dB.

Thus for sidelobe level control below -40 dB, the mainlobe width is doubled, and for control below -80 dB, it is quadrupled.

The widening of the filter response results in significant spectral overlap of adjacent filters. For example, filters with -70 dB sidelobes are down 3 dB at the center of the adjacent filter and are down 20 dB at the center of the filter removed two bins. Figure 2 demonstrates the wider mainlobe response at the DFT center frequencies.

FIGURE 1. Dirichlet Kernel DFT Filters

FIGURE 2. Three Term Blackman-harris DFT Filters

The classic response to the broadened filter response is to double the length of the filter to $2N$ points which of course reduces the bandwidth of the filter to $K(A) f_s/2N$. If the double length filter is now implemented in a double length transform the channel spacing is also reduced by a factor of two so it appears we are back where we

started. Actually, we have had a net gain. The sidelobe levels of the filters are now acceptable, the filter bandwidths are now correct, and we simply have too many filters.

We can perform the double length transform over the $2N$ point filter, obtain filter outputs which are spaced $f_s/2N$ apart and take alternate outputs to obtain spacings of f_s/N . In practice, the windowed data is circularly wrapped (or folded) from length $2N$ to length N and is presented to the N point transform. This intentional time domain aliasing is equivalent to subsampling the spectral points of the DFT. Equation 1 shows the equivalence of the two options and figure 3 demonstrates how the data wrapping is performed.

$$F(k) = \sum_{n=0}^{2N-1} f(n) w(n) e^{-j\frac{2\pi}{2N} nk} \quad \text{EQ.1a}$$

$$F(2k) = \sum_{n=0}^{2N-1} f(n) w(n) e^{-j\frac{2\pi}{N} nk} \quad \text{EQ.1b}$$

$$= \sum_{n=0}^{N-1} [f(n) w(n) + f(n+N) w(n+N)] e^{-j\frac{2\pi}{N} nk} \quad \text{EQ.1c}$$

FIGURE 3. Circularly Aliased Input Data

Thus the N narrowband channelized outputs are equivalent to those obtained from the alternate bins of a windowed and overlapped $2N$ point DFT. The problem we address is the reconstruction of the input data corresponding to the bandwidth spanned by a subset of these filter outputs. This is equivalent to forming the output series from a wider bandwidth filter. We synthesize this wider band by raising the sample rate and merging the time series from a collection of narrowband filters covering the same band of frequencies. The potential problem we have here is that only alternate bins of the double length transform have been formed. An alternative statement of the problem is that we have intentionally aliased the windowed data to avoid computing the redundant filters. We now have to undo that aliasing.

We argue that in forming the bank of filters spanning the spectral region of interest we have

not violated the Nyquist criterion relative to their individual bandwidths. As such, we have not lost any information by computing only the alternate spectral points of the double length DFT. Hence the missing spectral points are not required to undo the time domain folding and we should be able to perform the desired reconstruction without them.

THE CLASSICAL MERGING PROCESS

We can reconstruct the time domain signals corresponding to a selected group of narrowband filters by a technique known as inverse analysis. The spectral data was formed by an analysis process in which time data from overlapped intervals is projected onto a set of basis vectors which span the signal bandwidth. The structure of this processing is shown in figure 4.

FIGURE 4. Sequential Analysis and Synthesis

The $w_k(n)$'s are the impulse responses corresponding to the narrowband channels. These filters are nearly orthogonal and they are complete. Thus the reconstruction or synthesis can be performed as a simple weighted sum of the orthogonal basis vectors. This too is indicated in the synthesis section of figure 4. The merging of the reconstructions from the different overlapped intervals is also performed as a weighted summation. The merging process, shown in figure 4, is similar to the overlap and add fast convolution process.

The time series obtained from each of the narrowband filters is sampled at a rate which is commensurate with its low bandwidth. As the bandwidth of the composite signal is increased by the merging process, the sample rate of each contributing basis set vector must increase to reflect this higher bandwidth. It was an easy matter to discard samples from each series as we reduced their bandwidths but now we are faced with the task of plucking those samples back from the ether. We do this by imbedding an interpolator in the reconstruction process and we do it in a very simple way which is equivalent to the following description.

We know the center frequency of each basis vector. Rather than interpolate the values of the known vectors, we arrange to store a list of their samples. Then we increment through this list at the sample rate required to satisfy the bandwidth being assembled by the merging process. In fact we don't really do all of this work; after all we didn't explicitly have to compute the basis vectors for the decomposition, so we probably don't need to do so for the reconstruction. We synthesized the equivalent basis vectors in a sequence of operations; we first windowed the data (to account for the common magnitude of the basis vectors) and then used the DFT to supply the phase rotations for the vectors.

We again call upon the DFT to form the basis vectors. By loading the bins of the inverse DFT we automatically increase the output sample rate. More bins require larger transforms, and the larger transform will return more samples over the interval for which the sum of the basis vectors is being formed. Hence the IDFT supplies the basis vectors, the weighted summation of those vectors, and their interpolation to the higher sample rate required for the reconstruction.

The final problem, and the purpose of this paper is the minor detail that the basis vectors used for the decomposition were intentionally circularly aliased prior to the DFT and the IDFT (by itself) can not reconstruct them. In other words, each data point applied to the transform is the sum of two other points, and while the inverse transform can bring back that same summed point, it is not capable of pulling them back apart. This is closely related to the question, can we unscramble an egg?

MERGING WINDOWED AND FOLDED DATA

In the previous section we noted the problem yet to be solved is the data unfolding. We note that the DFT processes N points of data per interval of the form listed in equation 2.

$$h(n) = w(n) d(n) + w(n+N) d(n+N) \quad \text{EQ.2}$$

It is quite obvious we can not reconstruct $d(n+N)$ given $w(n)$, $w(n+N)$, and $h(n)$. But wait! The data $h(n)$ contributes to more than one observation. The full set of observations may contain the information to help us separate the folded data.

Before we look carefully at the data in the time domain, it is useful to examine its spectral description and in particular, how the spectral data we do have is related to the spectral data we don't have (ie. the missing bins). We denote the unwindowed spectral data as $H(k)$, the windowed spectral data as $G(k)$, and the window terms as $W(k)$. We note good windows have only a few non zero terms symmetrically about the zero index. The windowed terms for $k-1$, k , and $k+1$ for a two-term window (such as the Hamming or Hann) are shown in equation 3.

$$G(k-1) = W(0)H(k-1) + W(1)[H(k-2)+H(k)] \quad \text{EQ.3}$$

$$G(k) = W(0)H(k) + W(1)[H(k-1)+H(k+1)]$$

$$G(k+1) = W(0)H(k+1) + W(1)[H(k)+H(k+2)]$$

Suppose the k -th spectral point is one of the terms not computed by the windowed and folded transform and we wish to rebuild it. Note the data terms required to compute the k -th term are, along with other data, in the adjacent windowed bins. We can replace the missing k -th term with a weighted sum of the (non-missing) neighbors. Denoting the weights $A(k)$ and the weighted term as $G'(k)$ we have a representative form of the k -th spectral approximation shown in equation 4.

$$\begin{aligned} G'(k) &= A(1)[G(k-1)+G(k+1)] \\ &= A(1)W(1)H(k) \\ &\quad + A(1)W(0)[H(k-1)+H(k+1)] \\ &\quad + A(1)W(1)[H(k-2)+H(k+2)] \end{aligned} \quad \text{EQ.4}$$

The weighted sum of spectral neighbors appears to be some "other" spectral window. A window with the same number of spectral terms as the original input window might be a best bandwidth match for the spectral interpolator. We'll see shortly! Of course sliding the window through the spectra is the same as applying the window multiplicatively to the time series. The time series is the one formed with the missing spectral points set to zero. This is the same as taking the IDFT without the zeros and then replicating the resultant series. That's the way we will actually attempt the reconstruction. We take the spectra of interest through an IDFT, replicate it, rewindow it, and merge the overlapped intervals by straight summation.

We can predict the outcome of the reconstruction with the aid of a simple test, the impulse response of the whole processing chain (without the bandwidth reduction). We will apply the sequence of operations to a single data point at position "n" in the input array and note how well the impulse is reconstructed. Since we are only performing linear operations we will have characterized the entire process. The experiment is best followed with the aid of figure 5 which tracks a four-to-one overlapped and windowed DFT processor. The position of the non-zero input and output terms are noted at each step in the process.

FIGURE 5. Window And Fold Analysis Followed By Replicate And Window Synthesis

The single input data point at position n is denoted $h(n)$. It is presented to the transform as $g(n) = w(n)h(n) + w(n-N)h(n-N) = w(n)h(n)$. The point $g(n)$ has two possible parents which in general are not distinguishable but are in our controlled experiment. Upon returning through the inverse transform we place $g(n)$ at its two possible parent addresses (via the replicate) and apply the second window $v(n)$. Note that while the replicated data at the two locations is the same, the second window terms $v(n)$ and $v(n-N)$ are not. The two response positions and values are

$$r(n) = v(n)w(n)h(n) \text{ and} \\ r(n-N) = v(n-N)w(n)h(n)$$

If we repeat this operation for four overlapped intervals and then merge them by summing we obtain a reconstruction with non zero data at three positions as seen in figure 5. The center position is the expected reconstruction while the two artifacts at position $n-N$ and $n+N$ are pre-and-post echos.

In retrospect these echos are no surprise; the spectrum from which this data came has a periodic magnitude ripple (between the missing and present spectral data) with a period of 2 spectral bins (ie. 2 times $1/2N$ or $1/N$). It is easily shown, as in equation 5, that periodic ripple in the magnitude response of a system has the effect of generating a pair of pre and post echos about the main response offset by a distance equal to the reciprocal period N about the expected response.

$$G(\theta) = H(\theta) [1 + 2 \epsilon \cos(N\theta)] \quad \text{EQ.5}$$

$$= H(\theta) + \epsilon H(\theta) e^{+jN\theta} + \epsilon H(\theta) e^{-jN\theta}$$

$$g(n) = h(n) + \epsilon h(n+N) + \epsilon h(n-N)$$

Echo cancellation is a well posed problem in the communication sector and a simple zero forcing equalizer will attenuate the echos to any desired level. The Z-Transform of the system response is shown in equation 6. Also shown is the factored form of the same polynomial. Each factor corresponds to stable positive and negative time sequences.

$$H(Z) = \epsilon Z^{+N} + 1 + \epsilon Z^{-N} \quad \text{EQ.6}$$

$$= (a + bZ^{+N})(a + bZ^{-N})$$

$$\text{where } a = 0.5 + \sqrt{0.25 - \epsilon^2}$$

$$\text{and } a^2 + b^2 = 1$$

The Z-Transform of the zero forcing equalizer is the inverse of each of these factors and is shown, within a scale factor, in equation 7. Also shown is the expanded form of the two series.

$$G(Z) = \frac{1}{1 + cZ^{+N}} \frac{1}{1 + cZ^{-N}} ; c = \frac{b}{a} \quad \text{EQ.7}$$

$$= [\dots - c^2 Z^{2N} - cZ^N + 1][1 - cZ^{-N} + c^2 Z^{-2N} \dots]$$

Truncating the two series at L terms and introducing an L term delay to make the response realizable, we have the design of a $2L+1$ point zero

forcing equalizer.

The size of the main response and of the pre and post echos is position dependent and is related to the input and output window used for the analysis and synthesis. Figure 6 demonstrates how these terms vary with position (over $1/4$ -th of the interval) for a pair of 4-term Blackman-harris windows and for a 3-term B-h input window followed by a 4-term B-h output window. Thus the equalizer must use position dependent coefficients. These are of known form and can be drawn from a circulating memory as needed.

FIGURE 6. Ripples For Main And Pre & Post Echo

Figure 7 shows the residual levels of the pre and post echos for different combinations of input and output windows as a function of the truncated equalizer length. The window pairs used were various combinations of the Hann, and 3-term and 4-term Blackman-harris.

FIGURE 7. Residual Echo Level For Various Zero-Forcing Equalizers

FIGURE 8. Equalized Time Responses

Finally, figure 8 presents an example of the output responses for a set of zero forcing equalizers (of lengths 3, 5, and 7) when the input signal is a single short pulse. The processing used a Hann window and a 4-term B-h window to generate the spectral data and a 4-term B-h window to reconstruct the time data. The equalizer was approximated with the coefficient c (of equation 7) set to b/a (of equation 6), thus the zero forcing is inexact as seen in the -74 dB residue of the 3-tap equalizer. Even with this approximate coefficient, the equalization achieved with 5-taps exceeds -76 dB and with 7-taps exceeds -104 dB.

CONCLUSIONS

We have reviewed the classical technique for reconstructing time domain data from a sequence of overlapped and windowed DFTs. A common processing trick called data folding (or wraparound) decouples the length of the equivalent filter from the length of the DFT which performs the channelization. This is equivalent to desampling the output of a full length DFT and gives the impression that spectral data is discarded. Due to this impression there is a question whether the time data can be recovered from the desampled spectral points. We have demonstrated, that since the filter outputs we do compute completely span the desired spectra, we can reconstruct the time series with arbitrary fidelity. In response to a question we posed earlier, it appears that we can indeed unscramble an egg.

A NOTE OF THANKS

Special recognition is given to Kevin Knapp, a Graduate Student at SDSU (and an employee of Scientific Atlanta, San Diego) for programming the equalizers and for supplying figures 6 through 8. Kevin's Master's Thesis is a continuation of this work. Also thanks to Harry Nickerson and Ed Senior of GTE Sylvania for suggesting this problem.

REFERENCES

- Ronald E. Crochiere, and Lawrence R. Rabiner, "Multirate Digital Signal Processing", Prentice-Hall, 1983, Chapters 7.2.5 and 7.3.3
- John G. Proakis, "Digital Communications" McGraw-Hill, 1983, Section 6.4.