Comparing Occam and Wiener Filters on Broad-band Signals

Balas Kausik Natarajan and Konstantinos Konstantinides

Hewlett-Packard Laboratories
Computer Research Center
Palo Alto, CA

Abstract

Occam filters are a class of filters for additive random noise, based on the idea that when a lossy data compression algorithm is applied to a noisy signal with the allowed loss set equal to the noise strength, the loss and the noise tend to cancel rather than add. In this paper, we apply non-linear Occam filters to broad-band signals. Using the chirp signal as a specific example, we find that the Occam filter outperforms the Wiener filter consistently.

Key Words: compression, non-linear filtering, signal processing.

1. Introduction

Efficient filtering of signals corrupted with additive noise is central in many signal processing applications. Traditional spectral filtering techniques, such as Wiener filtering, often require some prior knowledge of the noise and signal characteristics. Here we consider a new class of filters that do not require such prior information but do not compromise on performance.

Consider a noisy signal, corrupted with additive random noise of known strength. The strength may be measured as, say, the amplitude or the power of the noise. Compress the noisy signal with a lossy data compression algorithm, with the loss allowed of the algorithm set equal to the strength of the noise. Will the loss and the noise add or will they cancel? It has been established [1], [2] that the loss tends to cancel the noise, with the extent of the cancellation depending on the compression achieved and how often the signal is sampled. This leads to the following technique, which is the essence of an Occam filter. *Compress the noisy signal with a lossy compression algorithm, with the allowed loss set equal to the noise strength. The decompressed signal is the filtered signal.* The technique is rather general, for instance, threshold filters using wavelet decompositions [3], [4], [5] can be viewed as special cases.

We consider the problem of filtering additive random noise from broad-band signals. Using compression algorithms that operate in terms of the piecewise linear functions, we construct Occam filters for the problem. We then compare the performance of the Occam filters with the Wiener filter, on a chirp signal corrupted with random noise. We find that the non-linear Occam filters offer better noise rejection than the Wiener filter.

2. Results

2.1 Preliminaries

Without loss of generality, we consider functions f on the unit interval. A sequence of n samples of f is a uniform sampling of f on the unit interval, i.e., $f_n = \{ f(0), f(1/n), f(2/n), \ldots \}$. Let ν be the random variable representing the noise. We use $\hat{f}_n$ to denote the sequence f_n corrupted with noise. A metric is a measure of the distance between two sequences. For two sequences f_n and g_n, the power metric is defined as

$$| f_n, g_n |_2 = \frac{1}{n} \sum_{i=0}^{n-1} (f(i/n) - g(i/n))^2 \ .$$

The power of a sequence f_n is its distance from the zero sequence, $| f_n, 0|_2$. The amplitude metric is defined as

$$| f_n, g_n |_\infty = \max_{i=0}^{n-1} | f(i/n) - g(i/n) | \ .$$

The amplitude of a sequence f_n is its distance from the zero sequence, $| f_n, 0|_\infty$.

With respect to a metric $| \cdot |$, a *lossy compression* algorithm C is a program that takes as input a sequence f_n and a loss tolerance $\varepsilon \geq 0$, and produces as output a binary string s representing a sequence g_n such that $| f_n, g_n | \leq \varepsilon$. C is said to obey the metric $| \cdot |$. A *decompression* algorithm D takes as input a binary string and produces as output a sample sequence. In particular, the decompression algorithm D corresponding to C would output g_n on input s. We use $C(f_n, \varepsilon)$ to denote the string s obtained by running C on input f_n and ε, and we use $D(s)$ to denote the sequence g_n obtained by running D on string s.

Using the above notation, we can state our filtering technique in the form of an algorithm. In the following, the strength of the noise is measured in the same metric as that obeyed by the compression algorithm.

Filtering Algorithm
input $\hat{f}_n$
begin
  Let $|\nu|$ be the strength of the noise.
  Run $C(\hat{f}_n, |\nu|)$;
  Decompress to obtain the filtered sequence g_n;
end

The filtering algorithm above requires the strength of the noise $|\nu|$ to be known. In this section we present a heuristic for estimating $|\nu|$. A more detailed discussion of this heuristic can be found in [2].

Calibration Algorithm
input $\hat{f}_n$
begin
  Run $C(\hat{f}_n, \varepsilon)$ for various values of
  ε, and plot output size versus $\log(\varepsilon)$;
  Let ε^* be the knee point of this plot, i.e the point
  at which its second derivative attains a maximum;
  Output ε^* as an estimate for the strength of the
  noise;
end

2.2 The piecewise linear representation

In this section we select a compression algorithm that operates in terms of the piecewise linear functions and obeys the L_∞ or amplitude metric. Using this compression algorithm we build an Occam filter and examine its properties.

The compression algorithm does the following. Given a sequence f_n and a tolerance ε, the algorithm constructs a piecewise linear function g such that $|g_n, f_n|_\infty \leq \varepsilon$, and g consists of the fewest number of pieces over all such piecewise linear functions. The output of the compression algorithm is the sequence of break points of the piecewise linear function g. Decompression is achieved by linear interpolation of the break points.

It happens that the compression scheme described above can be implemented optimally as an algorithm requiring time linear in the number of input points, using visibility techniques. Details can be found in the literature [6]. Also, a simplified form of the optimum algorithm that is amenable to hardware implementation is described in [7]. The simplified algorithm is not guaranteed to output a minimum number of pieces, but in practice we find that is within a factor of 1.5 of the minimum.

We use both the optimum and the simplified algorithms mentioned above to construct Occam filters as per the Calibration and Filtering Algorithm we gave earlier. We refer to the filter obtained from the optimum algorithm as Occam-O and to the simplified version as Occam-S.

2.3 A Broad-band signal

As an example of a broad-band signal, we select the function

$$f(x) = \begin{cases} 0 & x \leq 0.2 \\ \sin\left(\frac{1}{(x-0.2)+0.03}\right) & \text{otherwise .} \end{cases}$$

This signal has broad spectral support, and is difficult to filter with a classical spectral filter. We now add random noise to the function. We select the noise to be a uniformly distributed random variable in the range [-b,+b]. By selecting various values of b, the signal-to-noise ratio of the noise signal can be controlled. Figure 1 shows the 1000 samples of the function f, corrupted with noise generated by a pseudo-random number generator obeying the above

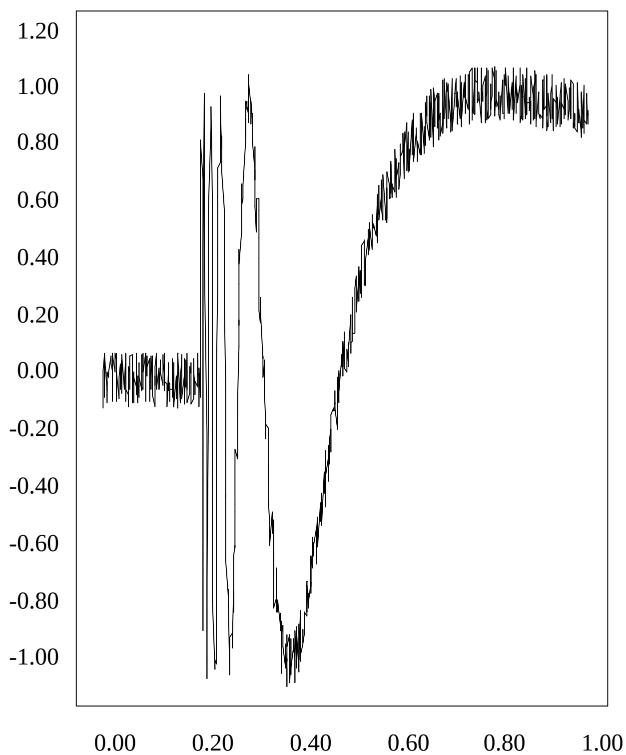

Fig. 1: A chirp signal corrupted with additive random noise uniformly distributed in the range [- 0.1,0.1]. SNR is 22dB.

distribution for $b = 0.1$.

2.4 The Wiener filter

The Wiener filter requires that the spectral properties of the noise-free signal and the noise be known in advance [8]. Its transfer function is given by

$$H(\omega) = \frac{S(\omega)}{S(\omega) + N(\omega)} .$$

where $S(\omega)$ is the power spectral density of the noise-free signal and $N(\omega)$ is the power spectral density of the noise. Since we assume the noise variable to be statistically independent at each sample point, it has uniform power spectral density and $N(\omega)$ is a constant that depends only on the variance of the noise distribution. We implement the Wiener filter using the discrete Fourier transform.

2.5 Performance

We now compare the performance of the Occam filters and the Wiener filter, on the chirp signal described earlier. First, we hold the sampling rate fixed at 1000 samples on the unit interval. We allow the noise strength to vary, selecting the amplitude b of the noise to take on the values 0.05, 0.1, 0.15, 0.2, ..., 0.4. Figure 2 shows a plot of the filtered sequence obtained by applying the Occam filter Occam-O on the noisy sequence of Figure 1, with the noise

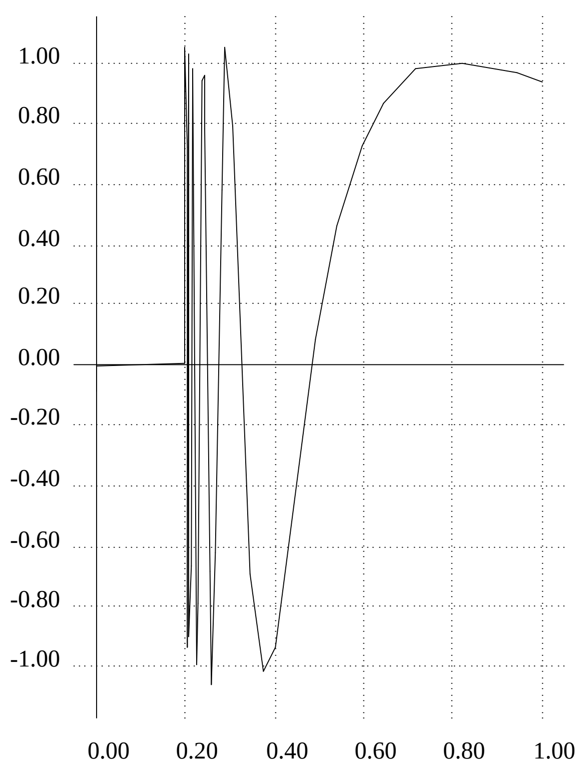

Fig. 2: Output of Occam-O filter filter given
the input of Fig. 1. Output SNR is 31 dB.

strength being estimated using the calibration algorithm. Figure 3 shows plots of the signal-to-noise ratio of the filtered sequence g_n against the signal-to-noise ratio of the noisy input

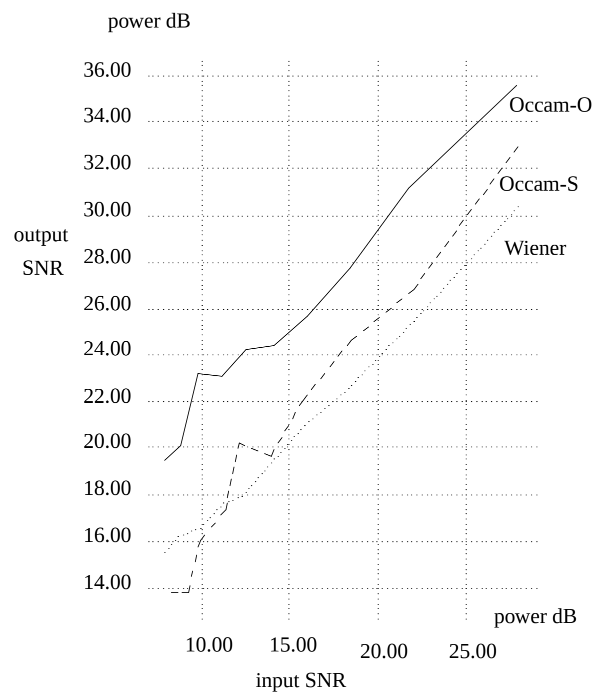

Fig. 3: Plot of SNR of filtered signal versus SNR of input signal.

sequence, both quantities being the power expressed in dB. Specifically, the signal-to-noise ratio of the filtered sequence g_n is

$$10 \log_{10} \left(\frac{| f_n |_2}{| f_n, g_n |_2} \right).$$

The signal-to-noise ratio of the input sequence $\hat{f}_n$ is

$$10 \log_{10} \left(\frac{| f_n |_2}{| f_n, \hat{f}_n |_2} \right).$$

From Figure 3, we observe that the Occam filter based on the optimum compression algorithm, Occam-O, performs best, followed by the Occam filter based on the simplified compression algorithm, Occam-S, followed by the Wiener filter. At the selected sampling rate, the Occam filter Occam-O consistently performs 5dB better than the Wiener filter over the input noise levels that we studied.

Next, fixing the noise strength at $b = 0.1$, we vary the sampling rate over 500, 1000, 1500, ..., 4000 samples on the unit interval. Figure 4 shows plots of the signal-to-noise ratio (power dB) of the filtered signal against the sampling rate. From Figure 4, we observe that the Occam filter based on the optimum compression algorithm, Occam-O, performs best, followed by the Occam filter based on the simplified compression algorithm, Occam-S, followed by the Wiener filter. The Occam filter Occam-O performs between 5 and 7dB better than the Wiener filter over the range of sampling rates studied.

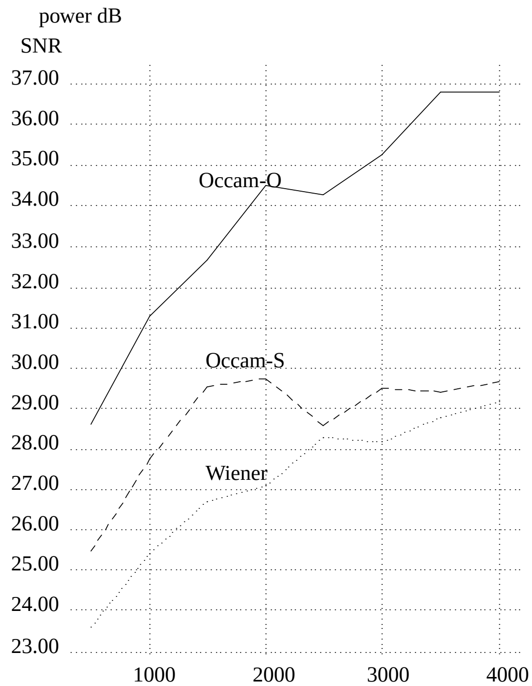

Fig. 4: Plot of SNR of filtered signal versus sampling rate, at input SNR of 22 dB.

3. Conclusion

We constructed two Occam filters based on compression algorithms that operate in the piecewise linear representations. We compared the performance of the Occam filters against that of a Wiener filter on a chirp signal corrupted with random noise. We found that the Occam filters consistently outperformed the Wiener. This is significant since unlike the Occam, the Wiener filter requires a priori information on the spectral properties of the noise and signal.

References

1. Natarajan, B.K., (1993). Filtering random noise via data compression, Proc. IEEE Data Compression Conference, Snowbird, Utah, pp.60-69.
2. Natarajan, B.K., (1994). A general technique for filtering random noise, Hewlett-Packard Laboratories Tech. Report HPL-94-55, June 1994.
3. DeVore, R.A., and Lucier, B. J., (1992) Fast wavelet techniques for near-optimal image processing, IEEE Military Comm. Conf., IEEE Comm. Society, 1992.
4. Donoho, D. L., Johnstone, I.M., Kerkyacharian, G., and Picard, D., (1993). Wavelet Shrinkage: Asymptopia?, Tech. Rep 419, Dept. of Statistics, Stanford University, Stanford, CA.
5. Saito, N., (1994). Simultaneous noise suppression and signal compression using a library of orthonormal bases and the minimum description length principle, Wavelets in Geophysics, to appear.

6. Imai, H., and Iri, M., (1986). An optimal algorithm for approximating a piecewise linear function. J. of Information Processing, Vol. 9, No. 3, pp. 159-162.

7. Konstantinides, K., and Natarajan, B.K., (1994). An architecture for lossy compression of waveforms using piecewise linear approximation, IEEE Transactions on Signal Processing, to appear.

8. Tretter, S.A., (1976) *Introduction to discrete-time signal processing,* John Wiley and Sons, New York, NY.