A Comparison of Correlation-Based Subsample Time Delay Estimation Methods for Periodic Signals

Travis Wiens[1,*], Stuart Bradley[2],

[1] Department of Mechanical Engineering
University of Saskatchewan
57 Campus Drive
Saskatoon, Sk
Canada
[2] Department of Physics
University of Auckland
38 Princes Street
Auckland
New Zealand

* Corresponding Author: t.wiens@usask.ca

Abstract

A common problem in radar, ultrasound, acoustics and other fields is that of determining the delay between two signals. In many situations, it is possible to determine the delay of a sampled signal to an accuracy better than the sample period. This paper presents a comparison of a number of correlation-based methods of estimating this subsample delay, with special attention to the case of periodic signals. The authors also suggest two new methods, which perform better than previous methods in some situations.

Introduction

A common problem in radar [1], medical imaging [2–5], seismology [6] and acoustic signal processing (among others) is determining the time delay between two signals . For example, this may be the delay between a signal's broadcast and the reception of its echo (radar) or the time of flight for an acoustic signal across a measurement area (acoustic tomography) [7].

In the simplest case, the delay can be estimated by finding the peak of the cross correlation between the two signals. In a discrete time (sampled) signal, the best resolution one can achieve for the delay using this method is equal to the sampling period. However, it is often possible to improve the delay estimation resolution by interpolating the cross correlation function between samples, using various methods [5, 8]. This paper compares the accuracy and computation time for a number of such subsample methods from the literature, as well as two improved methods. Particular attention is paid to the case of periodic signals.

The block diagram of a typical delay estimation system is shown in Fig. 1. A band-limited signal, $u(t)$, passes through two paths to two sensors. On each path is a filter, $G_0(s)$ and $G_1(s)$ (although they are often the same), including the dynamics of the sensors as well as the plant (e.g. the atmosphere in an acoustic problem). One path includes a pure time delay, t_d, which is assumed to be constant. Noise is added to each signal (here, we assume it is added after the filter). A sampling process returns the sampled discrete measurements, $x_0(k)$ and $x_1(k)$. It is assumed that the sampling rate, f_s, is greater than twice the maximum frequency of any signal component to avoid aliasing.

The problem is then to estimate t_d, given a set of measurements of $x_0(k)$ and $x_1(k)$.

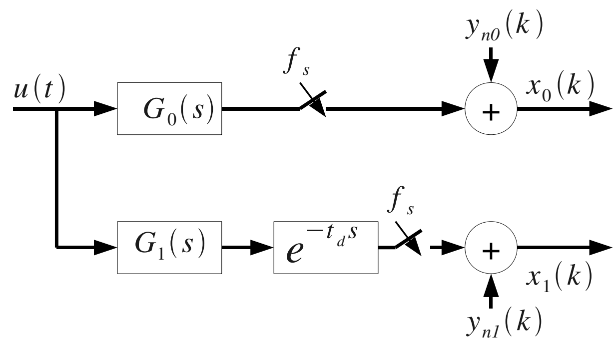

Figure 1. System block diagram, showing common source signal, filters, pure time delay, additive noise, and a sampling process.

Often the first step is to calculate the cross correlation of the measurements. For a periodic signal $u(t)$ with a period of N_p samples, the periodic discrete cross correlation between the sampled and noisy signals $x_0(k)$ and $x_1(k)$ is

$$xc_{01}(n) = \sum_{k=1}^{N_p} x_0(k)x_1(k+n) \tag{1}$$

where n is the lag. Circular convolution can be performed more efficiently using the Fast Fourier Transform, especially for large N_p:

$$XC_{01}(n) = X_0(n)X_1(n)^{\dagger} \tag{2}$$

where XC_{01}, X_0, and X_1 are the Discrete Fourier Transforms (calculated using the Fast Fourier Transform) of xc_{01}, x_0, and x_1, and † denotes the complex conjugate.

Delay Estimation Methods

Three Point Interpolation Methods

Since, for sampled measurements, the discrete cross correlation is only calculated at integer lags, it is necessary to interpolate the function in order to improve the estimation accuracy. A common method is to fit a parabola to three points: the peak of xc_{01} and its two neighbors [9, 10]. The peak of this parabola can then be found, indicating a subsample estimate of the delay. If the peak of the cross correlation is at $xc_{01}(k)$, the interpolated peak is at a delay of

$$\hat{t}_d = k + \frac{xc_{01}(k-1) - xc_{01}(k+1)}{2(xc_{01}(k-1) - 2xc_{01}(k) + xc_{01}(k+1))}. \tag{3}$$

Zhang [10] also presented the case for fitting a Gaussian curve to the three points surrounding the cross correlation peak, using

$$\widehat{xc}_{01}(t) = a\mathrm{e}^{-b(t-c)^2} \tag{4}$$

which has a peak at a delay of

$$\hat{t}_d = k + \frac{\ln(xc_{01}(k-1)) - \ln(xc_{01}(k+1))}{2(\ln(xc_{01}(k-1)) - 2\ln(xc_{01}(k)) + \ln(xc_{01}(k+1)))} \tag{5}$$

and is equivalent to fitting a parabola to the logarithm of the data.

A final method is to fit a cosine to the three points [11, 12]:

$$\widehat{xc}_{01}(t) = a\cos(\omega t + \phi). \tag{6}$$

The interpolated peak may be calculated using

$$\omega = \text{acos}\left(\frac{xc_{01}(k-1) + xc_{01}(k+1)}{2xc_{01}(k)}\right) \tag{7}$$

$$\phi = \text{atan}\left(\frac{xc_{01}(k-1) - xc_{01}(k+1)}{2xc_{01}(k)\sin(\omega)}\right) \tag{8}$$

$$\hat{t}_d = k - \frac{\phi}{\omega}. \tag{9}$$

Two Point Interpolation Method

Psarakis and Evangelidis [13] present a method of finding the subpixel peak of the cross-correlation of two images. This method can be used to determine the subsample delay between two signals with minimal modification. This method first normalizes the signals and removes the mean:

$$x_0^* = \frac{x_0 - \bar{x_0}}{\|x_0 - \bar{x_0}\|_2} \tag{10}$$

$$x_1^* = \frac{x_1 - \bar{x_1}}{\|x_1 - \bar{x_1}\|_2} \tag{11}$$

where $\bar{x}$ denotes the mean and $\|.\|_2$ denotes the Euclidean norm. The peak of the cross correlation (xc_{01}^*) of x_0^* and x_1^* is then found, as with the three-point method. This point and the larger of its two neighboring values are assumed to bracket the subsample peak on the interval between delays $k-1$ and k.

If one assumes that the signals are interpolated linearly between samples, a nonlinear equation for the cross correlation between peaks can be found, which has a maximum given by the analytical equation

$$\hat{t}_d = k + \frac{xc_{01}^*(k-1) - r\,xc_{01}^*(k)}{(r-1)(xc_{01}^*(k-1) + xc_{01}^*(k))} \tag{12}$$

where r is autocorrelation of x_0^* at a delay of one sample:

$$r = \sum_{k=1}^{N_p} x_0^*(k)x_0^*(k+1). \tag{13}$$

Phase Method

It is also possible to estimate the delay between the signals by taking advantage of the fact that a delay in the time domain is equivalent to a phase shift in the Fourier domain that is proportional to frequency [14]. As this linear phase shift can 'wrap' around 2π radians for large delays, which can make determining the slope difficult, this is implemented by first finding the integer peak of the cross correlation and shifting x_1 by this amount (i.e. aligning x_1 and x_0 to the nearest sample) and then using the phase to identify the subsample portion of the delay.

After aligning the signals to the nearest sample, the empirical transfer function estimate (ETFE) is calculated:

$$ETFE(\omega) = \frac{X_1(\omega)}{X_0(\omega)} \tag{14}$$

where ω is the angular frequency. One may then fit a line to the phase, ϕ, of the complex ETFE as a function of angular frequency, ω, and the delay estimate may be found from the slope:

$$\hat{t}_d = k - \frac{\mathrm{d}\phi}{\mathrm{d}\omega} \tag{15}$$

where k is the integer shift.

In order to improve noise-rejection properties of the estimate, one may weight the least squares estimate of the slope by the magnitude of the signal at each frequency (i.e. $|X_1(\omega)|$ or $|X_0(\omega)|$).

Improved Gaussian Method

One problem with the Gaussian method occurs if either of the points neighboring the cross correlation peak (i.e. $xc_{01}(k-1)$ or $xc_{01}(k+1)$) are negative. This will lead to calculating the logarithm of a negative number, causing a spurious complex result. While it is unlikely for this to occur with Zhang's test signals, it is possible for signals with large noise levels, or for signals with significant power above half the Nyquist frequency. Thus, we suggest a modified version of this method, which keeps the advantages of Zhang's method, but is more robust to the situation of negative cross correlations. This is achieved by adding a bias to the cross correlation before interpolation if there are any negative values, to ensure all values are positive. The bias is

$$\beta = -\min(xc_{01}(k-1), xc_{01}(k+1)) + \alpha \tag{16}$$

where α is a parameter with a real, positive value. The estimate for the delay is then

$$\hat{t}_d = k + \frac{\ln(xc_{01}(k-1)+\beta) - \ln(xc_{01}(k+1)+\beta)}{2(\ln(xc_{01}(k-1)+\beta) - 2\ln(xc_{01}(k)+\beta) + \ln(xc_{01}(k+1)+\beta))}. \tag{17}$$

It can be shown that when β=0, this algorithm is equivalent to Zhang's, and as β approaches infinity, the algorithm approaches the parabolic method. The optimal value for α depends on the specific signal under study, but it is not difficult to pick a value that outperforms Zhang's original method in a wide range of situations, as shown in the Results section.

Iterative Method

The three-point peak interpolation methods discussed in the previous sections rely on the fact that the cross correlation function can be approximated by a parabola, cosine or Gaussian curve. However, there will generally be some error in this approximation, especially when the delay is not near the integer and half-integer values [10]. In some situations, such as a very high signal to noise ratio, the error due to this misapproximation may be greater than the uncertainty due to the noise. In these cases, it may be advisable to attempt a more accurate subsample interpolation.

According to the Nyquist-Shannon Sampling Theorem [15], if a periodic signal is band-limited to a frequency of less than the Nyquist frequency, then the discrete Fourier transform of the signal contains enough information to interpolate the signals exactly. This also allows us to exactly calculate the cross-correlation function between integer samples. If we have the Fourier coefficients for the cross correlation, XC (calculated using equation 2), the continuous cross correlation function is

$$xc(t) = XC(0) + \sum_{n=1}^{N/2-2} 2XC(n)\mathrm{e}^{\frac{2\pi\mathrm{i}}{N_p}tn}. \tag{18}$$

Notice that, if N_p is even, we must discard the component at the Nyquist frequency (which we earlier assumed to be zero).

The peak of this continuous equation for $xc(t)$ is then found to estimate the delay:

$$\hat{t}_d = \mathrm{argmax}(xc(t)). \tag{19}$$

Unfortunately, no closed form solution exists, so numerical methods must be used to find the maximum. Brent's method with parabolic interpolation [16] works well, generally converging to a tolerance of less than 1×10^{-5} samples in less than nine iterations. Although this method of improving the delay estimation seems straight-forward, the authors are not aware of its existence in the literature, perhaps due to the requirement of a high signal-to-noise ratio for it to be worthwhile.

Other Methods

Other methods were considered for this analysis, but were not included in this paper. These include methods that solve the multiple-emitter problem (such as the MUSIC algorithm and its offshoots [17]), methods specific to image processing (e.g. [19]) and methods that track a moving signal (such as Derryberry and Dunham's subpixel search algorithm [18] or the adaptive least squares methods [20, 21]). However, as these methods are specifically designed to address certain problems that do not appear in this analysis, they can not be expected to be competitive with the methods presented here.

Performance Comparison Method

Zhang performed an accuracy comparison of parabolic and Gaussian delay estimation methods for a range of sample sizes and signal to noise ratios [10]. However, he only considered the case of a filter, $G(s)$ (equivalent to $G_0(s) = G_1(s)$ in Fig. 1), that resulted in an autocorrelation function of $xc(t) = xc(0)0.95^t$ which is approximately equivalent to a first-order Butterworth filter with cutoff frequency of 0.0163 times the Nyquist frequency. In this section we expand the comparison to include signals with different frequency content, as well as compare the new methods presented above. Additionally, computational benchmarks are presented to compare the computational cost of the various methods.

The first test examines the effect of signal to noise ratio (SNR) as well as the frequency content in the signal. An exactly white base signal was generated with a sample length of 256. This signal was then filtered by a second-order lowpass digital Butterworth filter with cutoff frequency f_c (filtering was computed by convolution of the signal with the filter's impulse response, performed in the frequency domain). This form of filter was selected, as it represents common physical systems, such as spring-mass-damper or capacitor-inductor-resistor systems, but the results are not particularly sensitive to the filter form or order. A delayed signal was then calculated by applying a linear phase shift to the filtered signal in the Fourier domain. Uncorrelated Gaussian white noise was then added to each signal. The delay between the filtered base signal and the delayed version was estimated using each of the methods discussed above. This was performed for 20 linearly spaced delays between -0.50 and 0.45 samples. At each point, 1000 repeats were performed with different randomly generated inputs and noise signals. The root mean squared error (RMSE) across the range of delays and for the 1000 repetitions was calculated. This was then repeated for all combinations of 100 linearly spaced signal to noise ratios between -5 and 25 dB and 100 filter cutoff frequencies that were logarithmically spaced between 0.01 and 0.99 of the Nyquist frequency, f_n. The uncertainty in each mean over the repetitions was calculated for 95% confidence.

The delay was estimated using algorithms for parabolic interpolation, Gaussian interpolation, the modified Gaussian interpolation, cosine interpolation, the two-point method, the linear phase method, and iterative interpolation. The tuning parameter, α for the modified Gaussian method was set to 1.0 (this value was selected somewhat arbitrarily, as a small positive value that ensured no negative logarithms). The iterative method was performed for 20 iterations or a tolerance of 1×10^{-5} samples, whichever occurred first.

Results

The results of the test are shown in Figures 2 through 8. Fig. 2 shows the root mean squared error, E_{best}, for the best method for each combination of filter cutoff frequency and SNR, averaged over the 1000 repetitions and 20 delay values. Figures 3 through 8 show the results for each method under discussion, as the ratio of the method's RMS error to the Ziv-Zakai lower bound, E_{zzb}. This bound is a theoretical lower bound for the RMS error, taking into account the frequency content of the signal and noise, and was calculated using Weiss and Weinstein's method [22, 23]. Also shown are confidence interval contours. The black line demarcates the area where the method is better than all others, with 95% confidence. The grey line marks where the method is as good as the best method, within the same confidence interval.

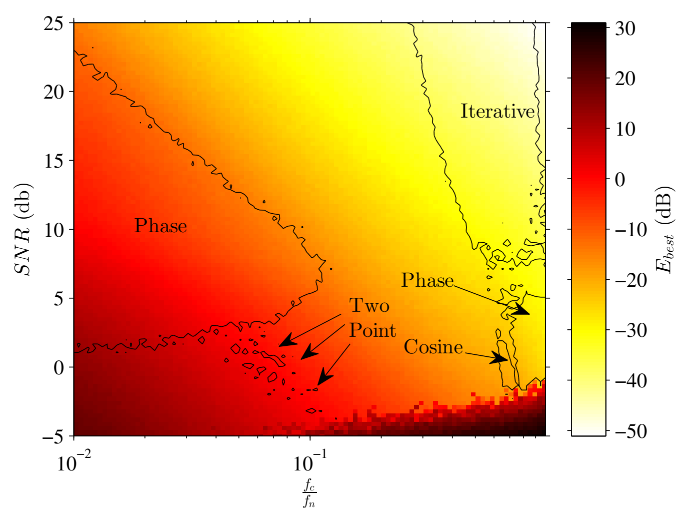

Figure 2. Effect of filter cutoff frequency and signal to noise ratio on delay estimation accuracy. RMS delay estimation error of the best method tested is shown by shading (on a logarithmic scale). The black lines show the area of the graph where the labeled method has a better mean error than any other, with 95% confidence.

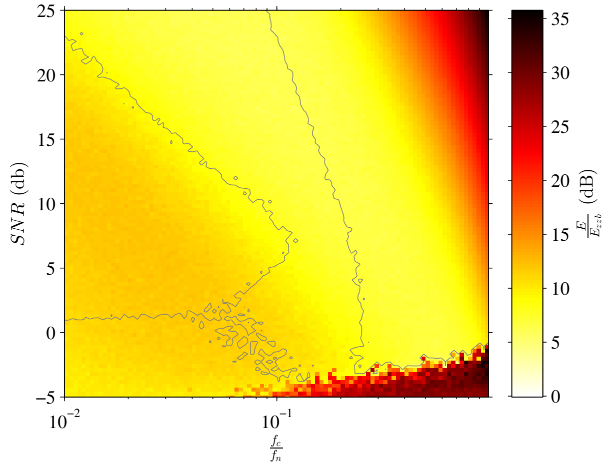

Figure 3. RMS delay estimation error for a parabolic interpolation, relative to the Ziv-Zakai lower bound. The grey lines show the area (toward the centre of the plot) where this method may be as good as the best, with 95% confidence. (There is no area of this figure where the parabolic interpolation method was significantly better than all others.)

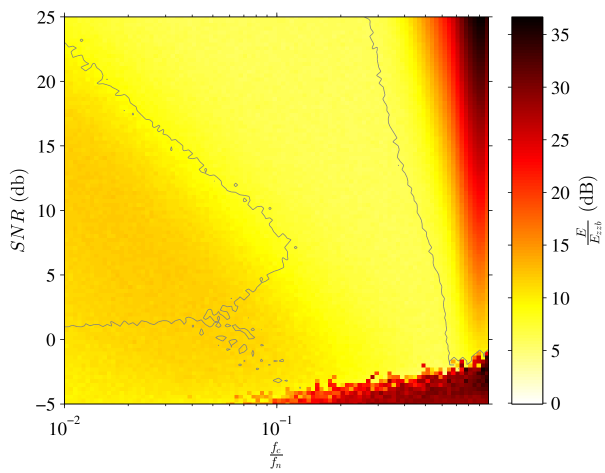

Figure 4. RMS delay estimation error for a Gaussian interpolation, relative to the Ziv-Zakai lower bound. The grey lines show the area where this method may be as good as the best, with 95% confidence. (There is no area of this figure where the Gaussian interpolation method was significantly better than all others.)

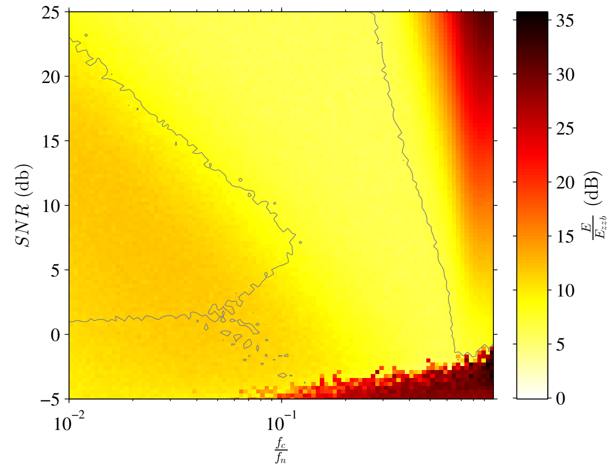

Figure 5. RMS error for the modified Gaussian interpolation, relative to the Ziv-Zakai lower bound. The grey lines show the area where this method may be as good as the best, with 95% confidence. (There is no area of this figure where the modified Gaussian interpolation method was significantly better than all others.)

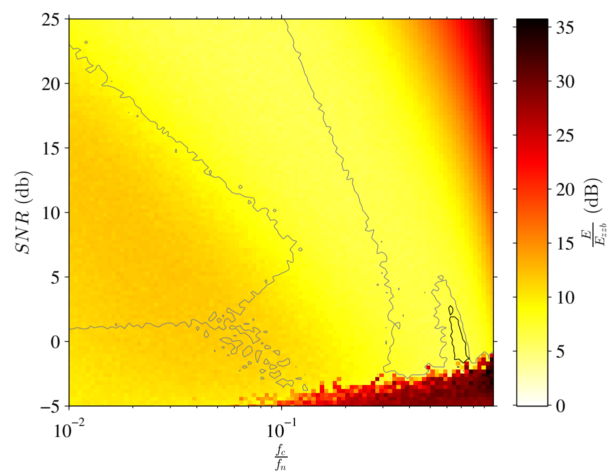

Figure 6. RMS error for the cosine interpolation, relative to the Ziv-Zakai lower bound. The black line shows the small area of the graph where this method has a better mean error than any other, with 95% confidence. The grey lines show the area where this method may be as good as the best, with 95% confidence.

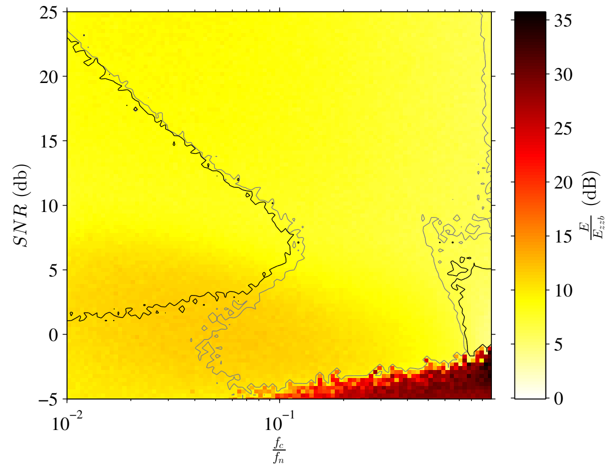

Figure 7. RMS error for the phase-based estimation method, relative to the Ziv-Zakai lower bound. The black lines show the areas of the graph where this method has a better mean error than any other, with 95% confidence. The grey lines show the areas where this method may be as good as the best, with 95% confidence.

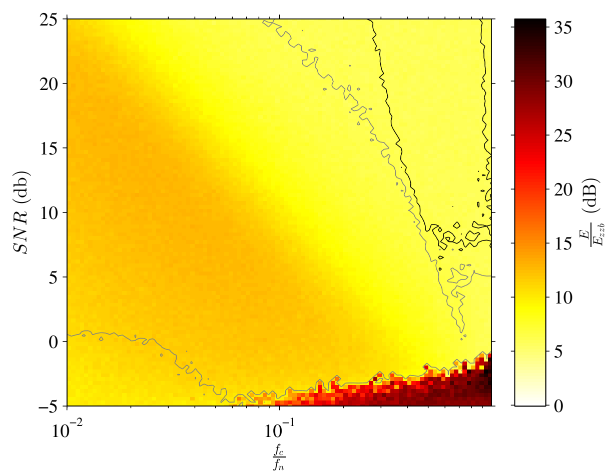

Figure 8. RMS error for the iterative estimation method, relative to the Ziv-Zakai lower bound. The black lines show the area of the graph where this method has a better mean error than any other, with 95% confidence. The grey lines show the area where this method may be as good as the best, with 95% confidence.

These results largely agree with Zhang's [10] for the Gaussian method and the cases he considered, although the additional results for higher cutoff frequencies show the strength of the other methods. The linear phase method is the best method for intermediate signal to noise ratios and low cutoff frequencies. The iterative method is an improvement over other methods for high SNR and high cutoff frequencies, although it is more sensitive to noise at lower cutoff frequencies. There are also small areas where the cosine and 2-point interpolation methods prevail. Also notice the large band in the middle of the plots where there is no significant difference between any of the three-point interpolation methods.

These results can be partially explained by the nature of the assumptions involved in each method. For high SNR and high frequency content the delay is less corrupted by noise, so the error in the subsample interpolation function becomes significant, which is most accurate for the iterative method. For signals with low frequency content, the peak of the cross-correlation function is wider and less pronounced. If the peak is considerably wider than two samples, the two- and three-point methods become imprecise as they are attempting to locate the peak from a small area of a relatively flat hill and are more easily corrupted by small errors. In contrast, the phase method is not restricted to this small subset of points, so it tends to excel in this situation.

Fig. 9 shows a comparison of the original Gaussian method and the modified version. The modified version is significantly better for cutoff frequencies above 0.6 time the Nyquist frequency and signal to noise ratios greater than 1, while outside of this area, there is no significant difference. The original method is not significantly better in any part of this plot.

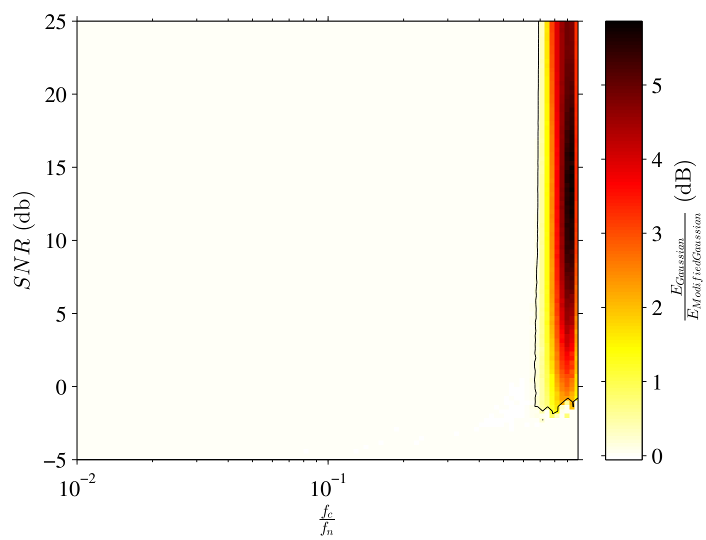

Figure 9. Comparison of delay estimation errors for Zhang's Gaussian interpolation method and the modified method. The black line shows the contour where the modified version's RMS error is better than the original method with 95% confidence. Outside of this area, the two methods are indistinguishable with the same confidence. There is no area on this plot where the original method is significantly better.

The computational time to perform the above test was also recorded. The code used is available as a supplement to this paper. It should be noted that, other than using the FFT to calculate the cross correlation functions, these algorithms have not been optimized and represent a naive implementation that should be able to be achieved by any researcher without difficulty. The program was run on Matlab 2013a on a single core of Intel Core i7 4770 CPU at a clock speed of 3.40 GHz with 24 GB RAM. Table 1 shows the mean CPU time required to estimate one delay value for each method. The cosine interpolation method is fastest by a small but statistically significant margin. It is interesting to note that even though the modified Gaussian method includes the additional step of adding a bias, it is faster than the original method, presumably because the original method requires the logarithm to be calculated using a complex algorithm, rather than scalar. The phase-based method takes slightly longer than the cosine method, but may be worthwhile due to its better accuracy in many cases.

Perhaps the most surprising result is the iterative method. In this test, the algorithm took an average of 8.63 iterations to reach a tolerance of 1×10^{-5} samples (with a standard deviation of 1.14 iterations), but the time required is only 4.5 times that of the fastest method. The improved accuracy (especially with a high filter cutoff frequency) may therefore not be as computationally expensive as one would have believed. It may also be prudent for users to examine the performance of an algorithm with only two or three iterations, as a compromise between accuracy and computation time.

Table 1. Mean calculation time for one delay estimation from two 256-point signals.

Method	Mean Time (μs)
Parabola	125.581 +/- 0.001
Gaussian	120.547 +/- 0.001
Modified Gaussian	118.632 +/- 0.001
Cosine	118.476 +/- 0.001
Two Point	198.847 +/- 0.001
Phase	125.145 +/- 0.001
Iterative	529.862 +/- 0.006

Conclusion

This paper has presented a comparison of subsample delay estimation algorithms, evaluated with a signal of varying frequency content and noise levels. It is apparent that the best choice of delay estimation algorithm is quite problem specific; for medium noise levels and low cutoff frequencies, the phase-based method was best; for low noise levels and high cutoff frequencies, the iterative method was most appropriate; while for areas in between, three-point peak interpolation schemes performed best. It should be noted, however, that there are large overlaps where it was not possible to differentiate between methods.

When computational speed is the prevailing factor, the three-point interpolation methods were best, with the cosine algorithm being fastest. While the phase and iterative methods were slower, it was somewhat surprising how little extra computational time these algorithms cost, especially when their superior accuracy in some situations is considered.

Two improved algorithms were also presented. A modified Gaussian interpolation method was introduced, which, for the same or less calculation time, guarantees no errors due to negative cross correlations (which can occur with large noise levels or a signal with significant power above half the Nyquist frequency). An iterative method of finding the peak of the cross correlation function was also presented, using Brent's method of optimization to find the peak of the continuous cross correlation function. Although this method did require significantly more computational time, it was also significantly more accurate for situations with a SNR greater than approximately 5 dB and the cutoff frequency greater than approximately half the Nyquist frequency.

References

1. I. Jameson. Time Delay Estimation. Report DSTO-TR-1705, Defense Science and Technology Organisation, Edinburgh, Australia (2006).
2. A. Eder, T. Arnold, C. Kargel. Performance Evaluation of Displacement Estimator for Real-Time Ultrasonic Strain and Blood Flow Imaging with Improved Spatial Resolution. IEE Trans. Instrumentation and Measurement 56;4 (2007) 1275-1284.
3. K. Hoyt, F. Forsberg, J. Ophir. Comparison of shift estimation strategies in spectral elastography. Ultrasonics 44 (2006) 99-108.
4. I. Cespedes, Y. Huand, J. Ophir, S. Spratt. Methods for Estimation of Subsample Time Delays of Digitized Echo Signals. Ultrasonic Imaging 17 (1995) 142-171.
5. F. Viola, W. F. Walker. A Comparison of the Performance of Time-Delay Estimators in Medical Ultrasound. IEEE Trans. Ultrasonics, Ferroelectrics, and Frequency Control 50;4 (2003) 392-401.
6. D. P. Schaff, F. Waldhauser. Waveform Cross-Correlation-Based Differential Travel-Time Measurements at the Northern California Seismic Network. Bulletin of the Seismological Society of America 95;6 (2005) 2446-2461.
7. T. Wiens, Sensing of Turbulent Flows Using Real-Time Acoustic Tomography. Proc. New Zealand Acoustic Society Conference (2008).
8. A. Fertner, A. Sjolund. Comparison of Various Time Delay Estimation Methods by Computer Simulation. IEEE Trans. Acoustics, Speech and Signal Processing 34;5 (1986) 1329-1330.
9. G. Jacovitti, G. Scarano. Discrete Time Techniques for Time Delay Estimation. IEEE Trans. Signal Processing 41;2 (1993) 525-533.
10. L. Zhang, X. Wu. On the application of cross correlation function to subsample discrete time delay estimation. Digital Signal Processing 16 (2006) 682-694.
11. P. G. M. de Jong, T. Arts, A. P. G. Hoeks, R. S. Reneman. Determination of Tissue Motion Velocity by Correlation Interpolation of Pulsed Ultrasonic Echo Signal. Ultrasonic Imaging 12 (1990) 85-98.
12. P. G. M. de Jong, T. Arts, A. P. G. Hoeks, R. S. Reneman. Experimental evaluation of the correlation interpolation technique to measure regional tissue velocity. Ultrasonic Imaging 13 (1991) 145-161.
13. E. Z. Psarakis and G. D. Evangelidis. "An enhanced correlation-based method for stereo correspondence with subpixel accuracy." Tenth IEEE International Conference on Computer Vision, Vol. 1. IEEE, (2005).
14. M. A. Rodriguez, R. H. Williams, T. J. Carlow. Signal Delay and Waveform Estimation Using Unwrapped Phase Averaging. IEEE Trans. Acoustics, Speech and Signal Processing 29;3 (1981) 508-513.

15. C. E. Shannon. Communication in the presence of noise. Proc. Institute of Radio Engineers, 37;1 (1949) 10-21.
16. W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery. Numerical Recipes - The Art of Scientific Computing, Third Edition. Cambridge University Press, New York, (2007).
17. A. M. Bruckstein, T. Shan, and T. Kailath. "The resolution of overlapping echos." Acoustics, Speech and Signal Processing, IEEE Transactions on 33.6 (1985): 1357-1367.
18. R.T. Derryberry and J.G. Dunham, "Search-based Subpixel Tracking." 1996 IEEE International Symposium on Circuits and Systems, (1996): 38-40.
19. S. K. Gehrig, H. Badino, and U. Franke. "Improving sub-pixel accuracy for long range stereo." Computer Vision and Image Understanding 116.1 (2012): 16-24.
20. S. Chern, and S. Lin. "An adaptive time delay estimation with direct computation formula." The Journal of the Acoustical Society of America 96.2 (1994): 811-820.
21. F. Reed, P. Feintuch, and N. Bershad. "Time delay estimation using the LMS adaptive filter–static behavior." Acoustics, Speech and Signal Processing, IEEE Transactions on 29.3 (1981): 561-571.
22. A. J. Weiss, E. Weinstein. Fundamental Limitations in Passive Time Delay Estimation - Part I: Narrow-Band Systems. IEEE Trans. Acoustics, Speech and Signal Processing 31;2 (1983) 472-486.
23. A. J. Weiss, E. Weinstein. Fundamental Limitations in Passive Time Delay Estimation - Part I: Wide-Band Systems. IEEE Trans. Acoustics, Speech and Signal Processing 32;5 (1984) 1064-1078.