

# Event Reconstruction by Digital Filtering

Olugboji Oluwafemi Ayodeji<sup>1</sup>, Jonathan Yisa Jiya<sup>2,\*</sup>, Jack M. Hale<sup>3</sup>

<sup>1</sup>Mechanical Eng. Dept. Federal University of Technology Minna

<sup>2</sup>Federal University of Technology, Minna, Niger state, Nigeria

<sup>3</sup>Department of Mechanical Engineering, Newcastle University, United Kingdom

\*Corresponding Author: jojileg@yahoo.com

Copyright © 2013 Horizon Research Publishing All rights reserved.

**Abstract** This work deals with a digital filtering technique that was developed to reconstruct a pulse after it has propagated along a pipe; a complex pulse that is progressively distorted. The technique developed makes use of the theory of digital filtering used in communications to remove distortion in long telephone links.

**Keywords** Filtering, Deconvolution, Pulse, Simulations, Fast Fourier Transforms, Reconstruction, Event, Signal

## 1. Introduction

Digital filtering is desirable in many situations in engineering and embedded systems. A good filtering algorithm can remove the noise while retaining the useful information. Many digital filters are based on the Fast Fourier transform (FFT), a mathematical algorithm that quickly extracts the frequency spectrum of a signal, allowing the spectrum to be manipulated (such as to create band-pass filters) before converting the modified spectrum back into a time-series signal using the inverse FFT. Two types of digital filters are discussed because of the relevance to this work: deconvolution filters and linear filters

### 1.1. Deconvolution Filter

Deconvolution filtering techniques are widely used in digital signal processing and image processing. They are of great significance in scientific and engineering applications. The technique, as developed in this research for pulse reconstruction, is based on the principle of the deconvolution filter used in communications to recover a distorted signal. In mathematics, deconvolution is referred to an algorithm-based process used to reverse the effects of convolution on recorded data [1].

In optics and imaging, deconvolution is applied in correcting the optical distortion that is associated with optical microscope, electron microscope, telescope and other imaging instruments. It is usually done in the digital domain by a software algorithm [2], as part of a set of techniques in

microscope image processing.

### 1.2. Linear Filters

All real measurements are disturbed by noise. This includes electronic noise, but can also include external events that affect the measurements taken, such as vibrations, variations of temperature, variations of humidity, etc., depending on what is measured and on the sensitivity of the device. It is often possible to reduce the noise by controlling the environment. Otherwise, when the characteristics of the noise are known and are different from the signals, it is possible to filter or to process the signal as mentioned above. Linear filters are useful in eliminating such unwanted frequencies (noise) from an input signal or to select a desired frequency among many other frequencies present in the signal.

## 2. Digital Filtering

This aspect deals with a technique developed for reconstructing the pressure pulse caused by an event such as an explosion or impact from measurements made by pressure sensors located along the pipeline. During the propagation of the pulse down the pipe it gets distorted in various ways before it reaches the place where it is being measured. It is necessary to reconstruct the original pulse before the distortion took place so that its potential for damage can be assessed. This is very much like the distortion removal in long distance telephone link by filtering.

Convolution is a formal mathematical operation, just as multiplication, addition and integration. Basically, addition takes two numbers and produces a third number while convolution takes two signals and produces a third signal. Supposing an input signal,  $x[n]$ , enters a linear system with an impulse response,  $h[n]$ , resulting in an output signal,  $y[n]$ . In equation form this can be written as;

$$x[n] * h[n] = y[n] \quad (1)$$

This is to say, the input signal convolved with the impulse response is equal to the output signal. Where \* represent

the convolution operator.

This forms the basis of digital filtering, since it allows a filter to be defined as a short time domain signal which is convolved with an incoming signal to give the required output in the same way as a conventional analogue filter acts on an analogue signal. It is therefore desirable to design a filter which when applied to the output signal will give a replica of the incoming signal. The filter should be capable of removing all the undesirable effects (distortion, noise, etc); such a filter is called a deconvolution filter.

The aim of deconvolution is to recreate the form of a signal as it existed before convoluting. This normally requires information about the characteristics of the convolution, such as impulse response or frequency response, to be known.

The application of deconvolution in signal processing is more straight-forward in the frequency domain than in the time domain, which allows computation to be faster [3]. The amplitude and phase of each sinusoid that comprises the original signal may be changed during the processes of convolution. To obtain the original signal, the deconvolution filter must undo these changes to the amplitude and phase of the signal. Smith [3] gave two examples to illustrate how the deconvolution filter can be applied to a signal, using a gamma ray detector and an old phonograph recording, respectively.

Generally, the objective of deconvolution is to find the solution of a convolution equation of the form:

$$f_i = h * f_o \quad (2)$$

where  $f_i$  is some recorded signal,  $f_o$  is the signal to be recovered, which has been convolved with some other signal  $h$  before it was recorded, and  $*$  is the convolution operator as mentioned earlier.

The function  $h$  might represent the transfer function of an instrument or a driving force that was applied to a physical system. If it is possible to know  $h$ , or at least know the form of  $h$ , then deterministic deconvolution can be performed on the signal. However, if  $h$  is not known in advance, then an estimate of it is required. This is most often done using methods of statistical estimation [4].

In physical measurements, the situation is usually closer to

$$f_i = h * f_o + \varepsilon \quad (3)$$

Where  $\varepsilon$  is noise that has entered the recorded signal. If it is assumed that a noisy signal or image is noiseless when trying to make a statistical estimate of  $h$ , the estimate will be incorrect, and so the estimate of  $f_o$  will also be incorrect. The lower the signal to noise ratio, the worse the estimate of the deconvolved signal will be. However, if it is possible to have some knowledge of the frequencies of the noise in the signal, it is possible to improve the estimate of  $f_o$  through filtering. Although signals are always delayed during the

passage through a filter, it is usually of no significance. The signal delay can be different for different frequencies [4], which mean that signals consisting of different frequency components suffer delay or time distortion.

## 2.1. Pulse Reconstruction by Deconvolution Filter

During the propagation of the pulse along the pipeline it gets distorted in various unknown ways before getting to the sensors where it is measured. This may be considered as the action of a filter. In DSP terms, this can be related as,

$$f_2(t) = h_k(t) * f_3(t) \quad (4)$$

where,  $f_2$ , is the measured pulse at sensor 2,  $f_3$ , is the measured pulse at sensor 3,  $h_k$ , is the required digital filter kernel, All these are discrete functions in the time domain, representing the true functions of time at a suitable sampling rate.

If  $h_k$  is known, then the original pulse at the event,  $f(E)$  can be reconstructed from the measured pulse signal  $f_2$ . This filter function  $h_k$  is obtained by the deconvolution of  $f_2$  and  $f_3$ .

This is actually difficult to do in the time domain, but fortunately very easy in the frequency domain [3]. It is another foundation of digital signal processing that convolution in the time domain is equivalent to multiplication in the frequency domain, so  $f_2 = h_k * f_3$  is the same as  $F_2 = H_k \times F_3$ , using the usual convention that an upper case function represents the Fourier transform (DFT) of the equivalent lower case time domain function. Then the deconvolution filter function in the frequency domain is simply obtained by the division  $H_k = F_2 / F_3$ , which is then transformed into a time domain function  $h_k$  using the inverse discrete Fourier transform.  $H_k$  is the frequency spectrum of the desired filter kernel,  $F_2$  and  $F_3$  are the frequency spectra of the measured pulses at sensors 2 and 3, respectively.

The solution to the above expression  $H_k = F_2 / F_3$  to determine  $H_k$  requires a complex division by the use of the magnitude and phase of  $F_2$  divided by the magnitude and phase of  $F_3$ . According to [71], this division in the frequency domain is achieved by an inverse operation of the multiplication of the two measured pulse signals at the sensors in the frequency domain by dividing their magnitudes and subtracting their phases. In polar form this is given by;

$$\text{Mag}(H_k) = \frac{\text{Mag}(F_2)}{\text{Mag}(F_3)} \quad (5)$$

$$\text{Phase}(H_k) = \text{phase}(F_2) - \text{phase}(F_3) \quad (6)$$

And in rectangular form it becomes,

$$\text{Re}(H_k) = \frac{\text{Re}(F_2)\text{Re}(F_3) + \text{Im}(F_2)\text{Im}(F_3)}{\text{Re}(F_3)^2 + \text{Im}(F_3)^2} \quad (7)$$

$$\text{Im}(H_k) = \frac{\text{Im}(F_2)\text{Re}(F_3) - \text{Re}(F_2)\text{Im}(F_3)}{\text{Re}(F_3)^2 + \text{Im}(F_3)^2} \quad (8)$$

The pulse at the start of the event can then be reconstructed using the convolution function,

$$f(E) = h_k * f_2 \quad (9)$$

where,  $f(E)$  is the reconstructed time domain pulse at the event site.

In situations where the propagation path affects the phase as well as the magnitude of the propagating pulse signal, some manipulations of the time domain function will be required so as to take into account the nature of the discrete Fourier transform which covers the frequency range  $\pm$  the Nyquist frequency and so transforms into an aliased time domain function. Extracting the deconvolution filter function under these conditions is described fully in [3].

The pulse reconstruction was implemented in an m-code program, using the FFT and convolution functions available within Matlab.

## 2.2. Pulse Propagation Model used to Develop Pulse Reconstruction Techniques

As an aid to the development of the various pulse reconstruction techniques investigated in this work, a standard model for a pulse propagating through a gas filled pipe was developed. The pulse takes the form of a sum of n

decaying sine waves, each at a frequency determined by ring modes of a pipe [5]. The pulse is subject to frequency dependent attenuation and dispersion as it propagates along the pipe.

Each component of the pulse is defined by; an exponentially decaying sine wave,  $e^{-\alpha t} \sin(2\pi ft)$ , where  $\alpha$  is a frequency dependent attenuation coefficient,  $f$  is the frequency and  $t$  is time measured from the pulse arrival; attenuation related to distance,  $e^{-\beta x}$  where  $\beta$  is a frequency dependent attenuation coefficient and  $x$  is the distance propagated; dispersion simulated by setting the start of each component of the pulse to a time defined by its frequency dependent phase velocity.

The attenuation coefficients  $\alpha$  and  $\beta$  are proportional to frequency squared [5].

The pulse is built up at each sensor location as a superposition of pulses, each at frequency of a ring mode and attenuation as defined above. The fundamental frequency of the pulse is 53 Hz, which comes from a pipe of Young's modulus of  $220 \times 10^9$  Pa, density of  $7860 \text{ kg/m}^3$ , wall thickness of 7.9mm and diameter 475mm. This fundamental frequency is obtained using equation (10) below;

$$K = 4 \left( \sqrt{\frac{EI}{uD^4}} \right)$$

where

E = Young's modulus (Pa)

$$I = \text{2nd moment of area (m}^4) = \frac{D \times t^3}{12}$$

D = Pipe diameter (m)

t = wall thickness (m)

$$u = \text{mass per unit length (k/m)} = \pi \times D \times t \times \rho$$

$$\rho = \text{density of pipe (kg/m}^3)$$

## 3. Application of Deconvolution Filter to Pulse Reconstruction

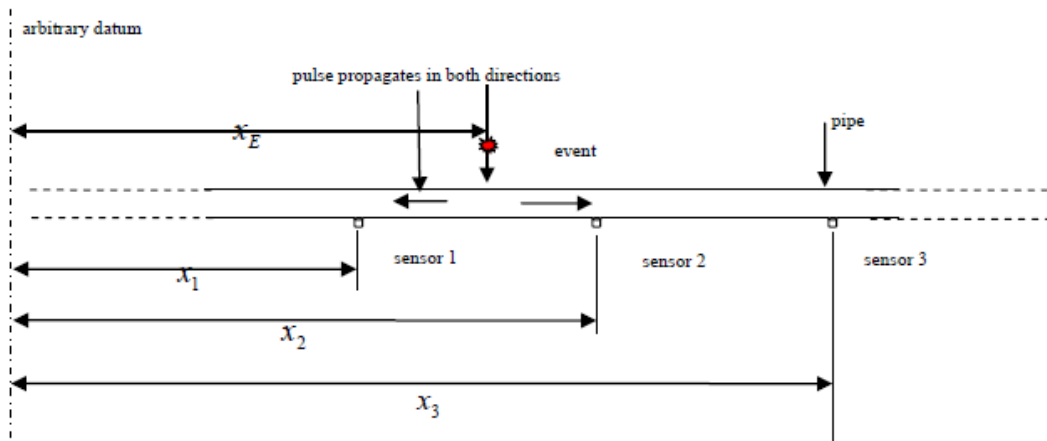


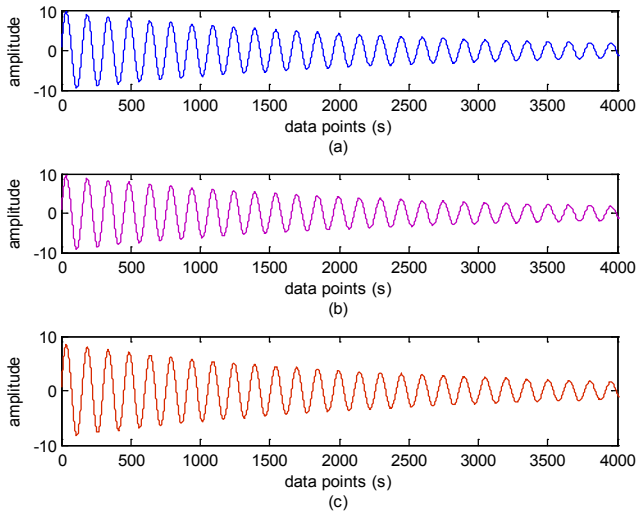
Figure 1. Schematic representation of sensors on a pipeline

### 3.1. Simple Pulse Reconstruction

This involved the use of the simplest form of pulse with a single frequency component, with a fundamental of the frequency of 53 Hz.

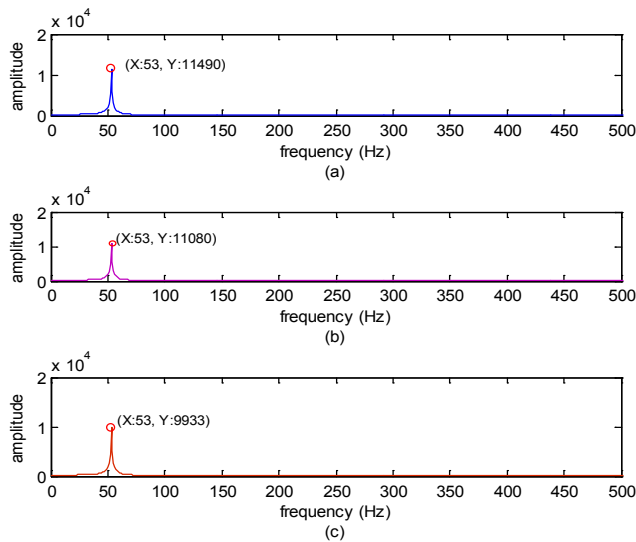
#### Case 1

Considering the first case with closely spaced sensors, Figure 1 shows the simulated pulses at the position of the event and that at sensors 2 and 3 in the time domain, while Figure 2 shows their frequency spectra.



(a) original pulse (b) pulse at 110 m (c) pulse at 410 m

**Figure 2.** Pulse responses obtained at the event and sensors 2 and 3 (single frequency with sensors close to event) in the time domain



(a) original pulse (b) pulse at 110 m (c) pulse at 410 m

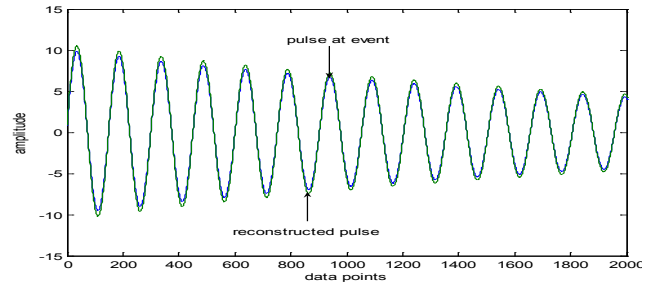
**Figure 3.** Pulse responses obtained at the event and sensors 2 and 3 (single frequency with sensors close to event) in the frequency domain

From Figures 1 and 2 the magnitudes of the pulses can be seen to be reducing gradually as the pulse propagates away from the event both in the time and frequency domain. This can be seen in Figure 2 that at the fundamental frequency of 53 Hz, the pulse at the event as shown in Figure 2(a) had reduced by about 14% in magnitude by the time it

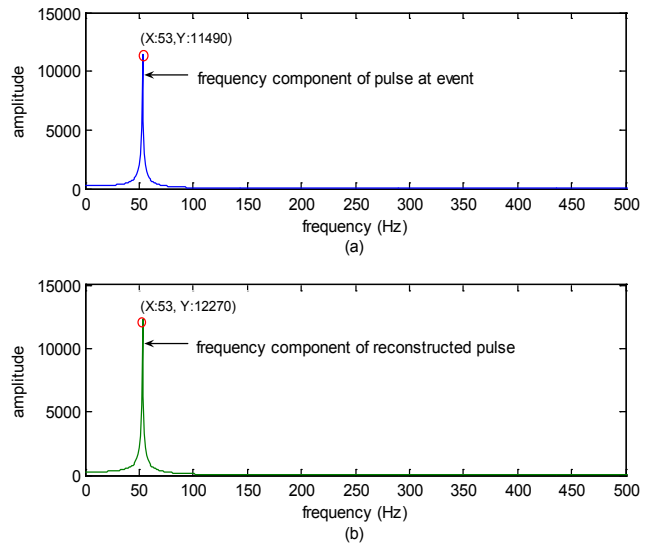
arrived at sensor 3 as seen in Figure 2(c).

This slight reduction in magnitude is as a result of the closeness of the sensors to each other and to the event. The single frequency peak at 53Hz can be seen in the frequency domain plot in Figure 2 at the event and both sensors.

The digital filtering technique, implemented in the m-code program as outlined in section 3, was used to obtain a deconvolution filter using the pulse signals obtained at sensors 2 and 3, which was then applied to the signal from sensor 2 to reconstruct the form of the pulse at the event site. Figure 4 and Figure 5 shows the original and reconstructed pulse in the time and frequency domains.



**Figure 4.** Original and reconstructed pulse at event location by deconvolution filter (single frequency with sensors close to the event)



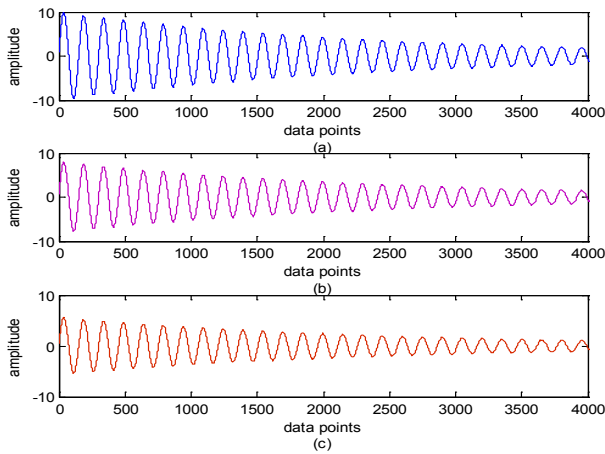
(a) original pulse (b) reconstructed pulse

**Figure 5.** Reconstructed simulated pulse at event by deconvolution filter (single frequency with sensors close to the event) in frequency domain

The form of the reconstructed pulse as shown in Figures 4 and 5 indicates a fair approximation to the original simulated pulse at the event location. The frequency, phase, amplitude and decay of the pulses all coincide well in Figure 4. This can be seen more clearly in Figure 5 where the original and reconstructed have the same frequency of 53 Hz and differ by only 7% in magnitude.

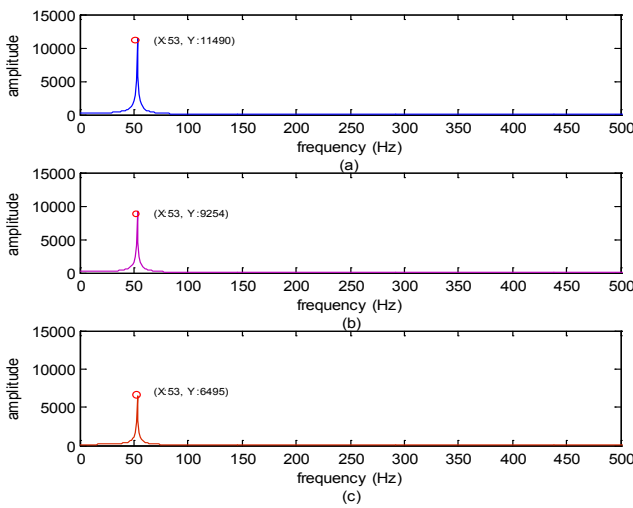
#### Case 2

For the second case with widely spaced sensors, Figures 6 and 7 show the simulated pulses in the time and frequency domains, respectively.



(a) original pulse (b) pulse at 800 m (c) pulse at 1800 m

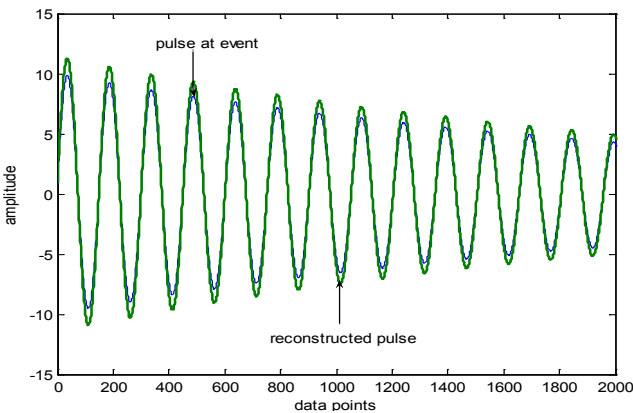
**Figure 6.** Simulated pulses with sensors far apart to the event (single frequency) in time domain



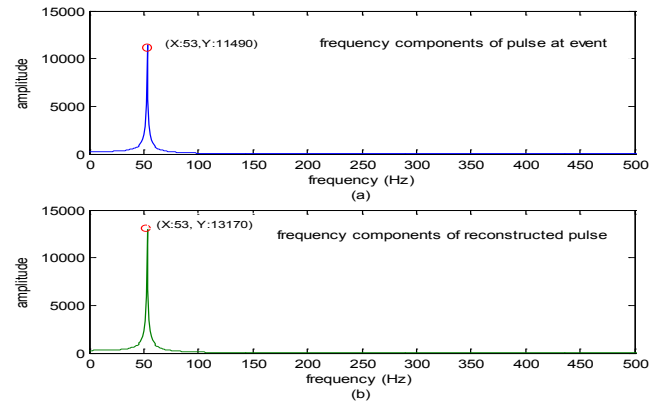
(a) original pulse (b) pulse at 800 m (c) pulse at 1800 m

**Figure 7.** Simulated pulses with sensors far apart to the event (single frequency) in frequency domain As before, the frequency of the pulses is unchanged, but the magnitudes are lower, reflecting the greater distance travelled.

Figures 8 and 9 show the reconstructed pulse and the original simulated pulse in the time and frequency domains.



**Figure 8.** Reconstructed simulated pulse at event by deconvolution filter (single frequency with sensors far apart to the event) in time domain



(a) original pulse (b) reconstructed pulse

**Figure 9.** Reconstructed simulated pulse at event by deconvolution filter (single frequency with sensors far apart to the event) in frequency domain

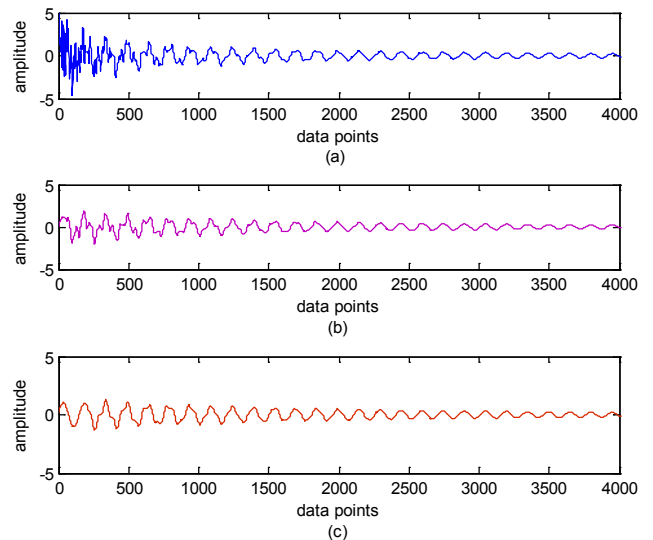
The reconstructed pulse shows similar characteristics to that obtained with the closer spaced sensors (case 1), except for the magnitude of the reconstructed pulse which is overestimated by 15%.

### 3.2. Complex Pulse Reconstruction

The ability of the digital filter technique to reconstruct a complex pulse was tested using a pulse with eight frequency components. As before, the two cases of pulse reconstruction with closely and widely spaced sensors are used.

#### Case 1

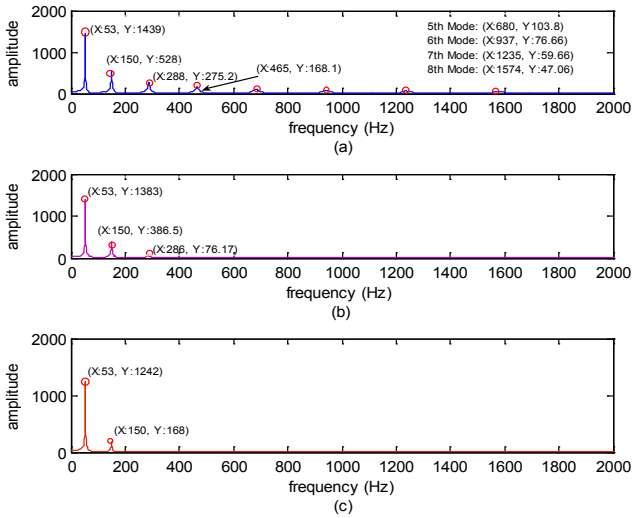
Figures 10 and 11 show the simulated pulses obtained with closely spaced sensors in the time and frequency domains.



**Figure 10.** Simulated pulses with sensors close to the event (multiple frequencies) in the time domain

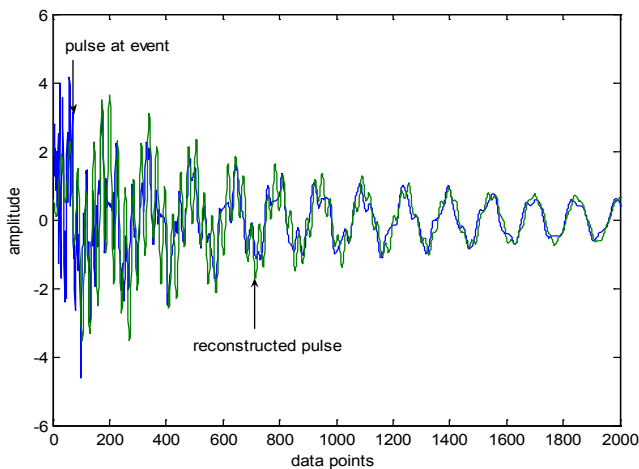
The form of the multi-component pulse may be seen in Figure 10(a), which is calculated without any attenuation. The high frequency components are clearly apparent at the start of the pulse, but decay quickly leaving only the low frequency components in the tail.

The simulated pulses at the sensor positions shown in Figures 10(b) and (c) show the effect of the frequency dependent attenuation with propagation. The higher frequency components of the pulse at the start of the event can be seen to be progressively reduced as the pulse arrives at sensors 2 and 3 and as before to diminish rapidly within the pulse. The effect of dispersion can also be seen in Figure 10(c), where a harmonic appears after one cycle of the fundamental component.



(a) original pulse (b) pulse at 110 m (c) pulse at 410 m  
**Figure 11.** Simulated pulses with sensors close to the event (multiple frequencies) in the frequency domain

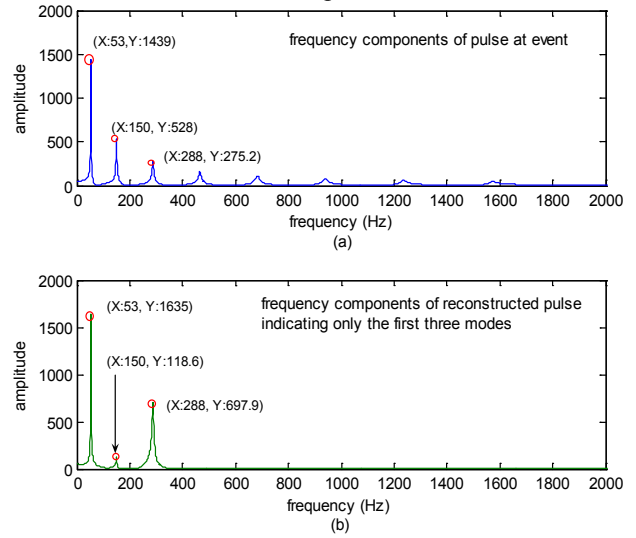
The frequency dependent attenuation can be seen more clearly in Figure 11. The magnitude of the fundamental component at 53 Hz decays slowly as before, but the higher frequency components decay much more rapidly so that the third and subsequent components cannot be seen in the linear plot used. Figures 12 and 13 shows the reconstructed pulse in the time and frequency domains.



**Figure 12.** Reconstructed simulated pulse at event by deconvolution filter (multiple frequencies with sensors closer to the event) in time domain

Figure 12 shows that with closely spaced sensors, it was still possible to reconstruct the pulse in a form that looks

similar to the original. The magnitude and decay rates are similar and the high frequency content is similarly concentrated at the start of the pulse.

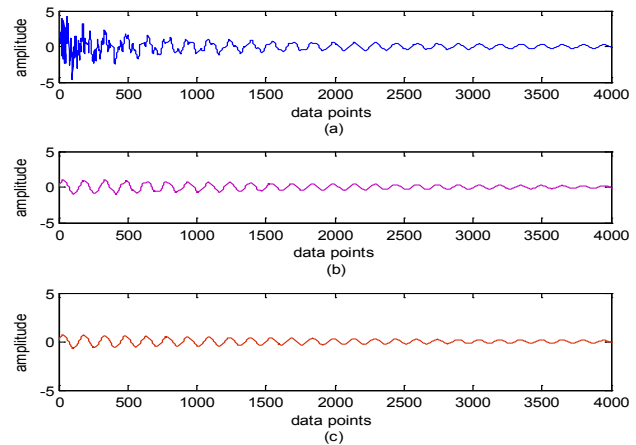


(a) original pulse (b) reconstructed pulse  
**Figure 13.** Reconstructed simulated pulse at event by deconvolution filter (multiple frequencies with sensors closer to the event) in frequency domain

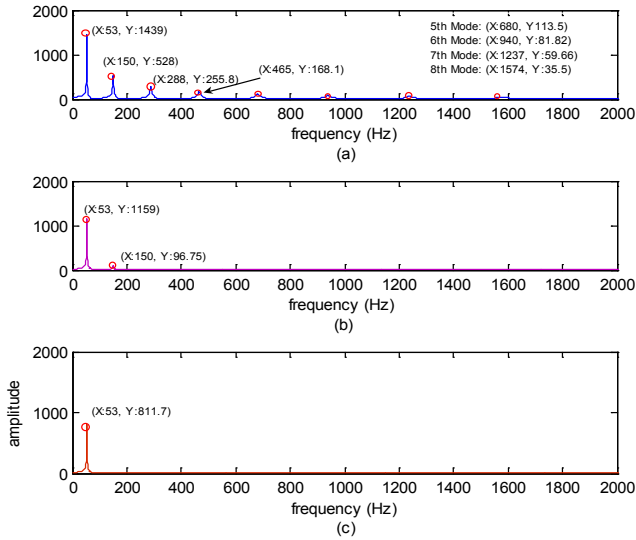
Figure 13 shows that this appearance is not entirely correct. The fundamental component is close, having the correct fundamental frequency of 53 Hz with a difference of 14% in their magnitudes. The second and third modes also have the correct frequencies of 150 Hz and 288 Hz, but differ in their magnitude by 78% and 60% in opposite directions; i.e. with the second mode being underestimated and the third overestimated. The higher modes are not reconstructed at all, presumably because they attenuated to such a high degree before reaching sensors 2 and 3 that they cannot be seen in Figure 11, and so would not have contributed to the deconvolution filter.

Case 2

For the widely separated sensors, Figures 14 and 15 shows the simulated pulses obtained at the event and sensors in the time and frequency domains, respectively.



(a) original pulse (b) pulse at 800 m (c) pulse at 1800 m  
**Figure 14.** Simulated pulses with sensors far apart to the event (multiple frequencies) in time domain



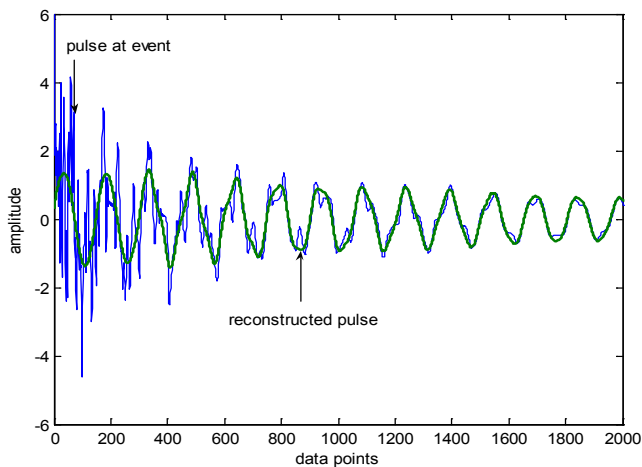
(a) original pulse (b) pulse at 800 m (c) pulse at 1800 m

**Figure 15.** Simulated pulses with sensors far apart to the event (multiple frequencies) in frequency domain

The high frequency content seen clearly at the start of the original pulse in Figure 14(a) is greatly reduced by the time the pulse arrives at sensor 2 and is practically invisible at sensor 3. This is shown more clearly in the frequency domain in Figure 15, where only the first mode peaks are clearly evident, but the second mode peak is barely visible at sensor 2 only and higher mode peaks cannot be seen at all.

Because of this the form of the reconstructed pulse as shown in Figure 16 below was affected, it was virtually difficult to see any of the high frequency components in the reconstructed pulse. This was confirmed from the frequency domain plot in Figure 17 where only two modes can be seen in the reconstructed pulse in Figure 17b compared to the eight modes at the start of the event in Figure 17a.

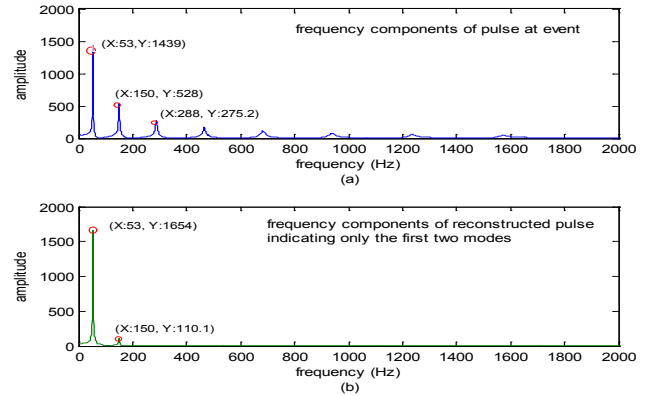
Figures 16 and 17 show the original and reconstructed pulses obtained from using the widely separated sensors.



**Figure 16.** Reconstructed simulated pulse at event by deconvolution filter (multiple frequencies with sensors far apart to the event) in time domain

Figure 16 shows that with the sensors widely separated and distant from the event, it is still possible to reconstruct a

pulse by deconvolution filtering, though it is not as good as that obtained with the sensors closely spaced. The magnitude, phase and decay rate of the first mode appear to be correct, but there is little sign of any higher modes at all.



(a) original pulse (b) reconstructed pulse

**Figure 17.** Reconstructed simulated pulse at event by deconvolution filter (multiple frequencies with sensors far apart to the event) in frequency domain

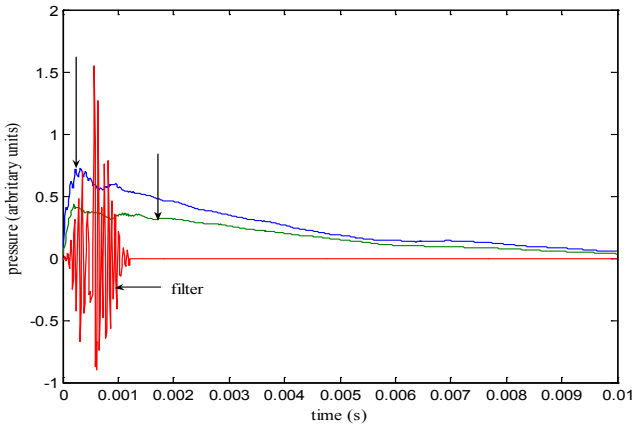
This is confirmed in Figure 17, which shows the spectrum of the reconstructed pulse with a peak at the correct fundamental frequency of 53 Hz, but with magnitude overestimated by 14%. The peak for the second mode is at the correct frequency of 150 Hz but is greatly underestimated in magnitude, and higher modes are not present at all. This enforces the intuitive expectation that the more closely the sensors are separated, the more accurately the pulse will be reconstructed.

These simulated test results obtained using the pulse propagation model show that the digital filtering method of pulse reconstruction will work for all measurable pulses with any sensor spacing provided the spacing is not so great as to make the pulse arriving at the more remote sensor too small to trigger the pulse capture. However, they show clearly the detrimental effect of wide spacing on the quality of pulse reconstruction. For practical applications a trade off must be made between the competing desires to reconstruct accurately and to make low cost systems. The simulation used here could be used as a design tool, using experimentally determined values for the attenuation and dispersion coefficients for the specific application. It will enable a pipeline system designer to establish suitable sensor spacing to give only the quality of reconstruction required to assess damage envisaged for that pipeline.

### 4. Experimental Work and Results

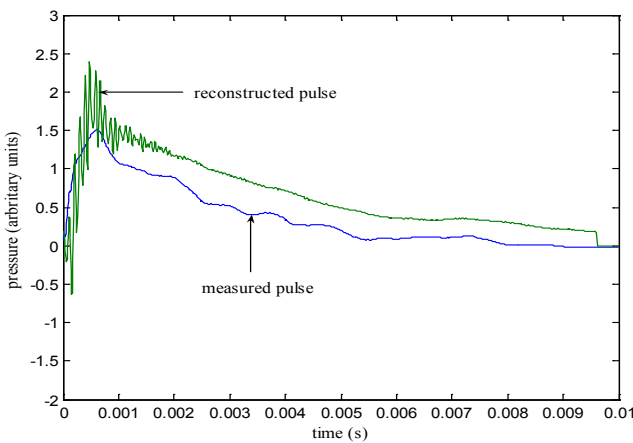
To test the effectiveness of the event reconstruction by digital filtering, the extracted pulse measurements  $f_2$  and  $f_3$  were used. The discrete Fourier transform of the pulses were calculated using the fast Fourier transform (FFT) function in MatLab, from which the discrete deconvolution function  $h$  was calculated. Figure 18 shows a

representative measurement of the pulse functions  $f_2$  and  $f_3$  from which the deconvolution function  $h$  was obtained, truncated at 100 data points and padded to 830 with zeros.



**Figure 18.** Filter function obtained from pulse measurements at sensors 2 and 3

As can be seen, there is no apparent relationship between the measured pulse at sensors 2 and 3 and the deconvolution function obtained from them. The filter could not have been obtained except by means of the transformation into the frequency domain as earlier described. The pulse function at the position of sensor 1 was then reconstructed using the convolution  $f_1(R) = h * f_2$  and scaled according to the different distances between the sensors. The reconstructed pulse is shown in Figure 19, compared with the true function  $f_1$  measured by the sensor.



**Figure 19.** Typical Reconstructed pulse at sensor 1 using the deconvolution filter method

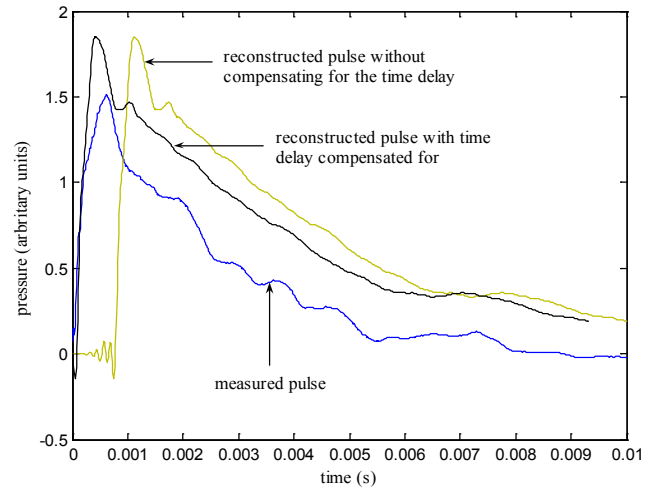
The shape of the reconstructed pulse is broadly similar, but is distorted by high frequency noise at the start of the pulse. Three reasons may be attributed to this discrepancy: the calculated deconvolution filter function is a finite length approximation to the true filter it includes the noise on the signals  $f_2$  and  $f_3$

The technique neglects the non-linearities inherent in the

propagation.

The problem of the filter can be reduced to some extent by increasing its length, though the improvement quickly reduces with further lengthening and is inherent in the use of the digital filters. The noise problem is also inevitable, particularly in a pipeline with flowing gas and when the sensors are well separated to keep down costs, making the signal at the most remote sensor fairly small.

This noise was easily removed by means of a low pass filter. A linear phase Blackman window filter of 100 points was found to work well as shown in Figure 20.



**Figure 20.** Reconstructed pulse signal at sensor 1 with low pass filtering

The form of the final reconstructed pulse shown in Figure 20 is a fair approximation to the true measured pulse at sensor 1. There are two main discrepancies: the peak magnitude is overestimated by about 17%, and the pulse shape is different in detail, most notably the introduction of an anomalous second peak. The delay which can be seen in the reconstructed pulse before it was compensated for is a normal effect of filtering and as such, is of no significance. This is evident from the reconstructed pulse when the delay was compensated for; moreover, the purpose of the reconstruction is to determine the pulse size and shape which carries the information required about cause of the event.

Also, the small oscillations before the rise is an artefact of the Blackman window used in the low pass filter. These results are for one out of the fifteen repeat tests and are typical. The range of magnitude overestimation was between 17 % and 23 % compared to the measured true pulse at sensor 1.

## 5. Conclusions

A Method of reconstructing the form of a pressure pulse at the site of an event causing it was developed using digital filtering technique. This was tested initially using a computer based modelled pulse propagation, and subsequently validated experimentally. The results obtained showed its



suitability to reconstruct the form of pressure pulse propagating along a gas filled pipeline from its source.

---

## REFERENCES

- [1] O'Haver T. "Intro to Signal Processing - Deconvolution". University of Maryland at College Park, 2007.
- [2] Cheng, P.C. "The Contrast Formation in Optical Microscopy". Handbook of Biological Confocal Microscopy (Pawley JB, ed.) (3rd ed. ed.). Berlin: Springer, pp. 189–190, 2006.
- [3] Smith, S.W. "The Scientist and Engineer's Guide to Digital Signal Processing". California Technical Publishing, 1997.
- [4] Wiener .N. (1964). "Extrapolation, Interpolation, and Smoothing of Stationary Time Series". Cambridge, Mass: MIT Press, 1964.
- [5] Cao, C.F, Baik, S., Choi, J.B and Kim, Y.J. "Protection of Underground Gas Pipelines from Third Party Damage by On-Line Monitoring Using Piezoelectric Accelerometers". Proceedings of the Institution of Mechanical Engineers, Part E:Journal of Process Mechanical Engineering. Volume 221, pages 61 – 67.