Deconvolution problem and application to ANITA signals

Debdeep Bhattacharya

November 8, 2018

Abstract

We discuss the mathematical problem of deconvolution with additive white noise and its application to time-localized signals observed by ANITA. Considering a simulated signal and the WAIS signals recorded by ANITA antennas, we study the classical Fourier based deconvolution methods, as well as relatively recent wavelet-based deconvolution ForWaRD algorithm. We also consider a multi-antenna model that represents the physical problem more accurately and compare its output with that of the single-antenna model.

Contents

1 Introduction

In this report, we outline several Fourier-based and one wavelet-based method to deconvolve time-localized signals observed by ANITA antennas. Using a simulated signal and a set of time-localized pulses (referred here as WAIS pulses) recorded by ANITA antennas, we conclude that the wavelet based deconvolution method performs better than Fourier-based methods, in terms of both relative error and signal-to-noise ratio.

This report is organised as follows. In Section 1, we fix notation and some definitions used throughout this report. In Section 2, we discuss the Fourier bases deconvolution techniques applied to the theoretical signal and ANITA signals. In Section 3, we introduce wavelet bases and discuss a wavelet-based deconvolution method called ForWaRD algorithm. In Section 4, we introduce a multi-antenna model to closely represent the mathematical problem related to deconvolving ANITA signals. In Section 5, we summarize our observations, difficulties and further directions.

The plots are generated with GNU/Octave. The octave functions (compatible with MATLAB syntax) used to generate the plots along with the documentation can be found here: `https://github.com/debdeep777/numerical/tree/master/wavelet`. Also, the main algorithm for the single-antenna model along with wavelet-related tools have been ported to C++ as a library and can be applied to a vector of datatype *double** to compute the deconvolution. The C++ library *WTools* can be found here: `https://github.com/debdeep777/libWTools`

1.1 Notation

Signals are represented using lower case letters with temporal variable t. For example, $x(t)$, $y(t)$ are signals, where $t = 0, \ldots, N-1$ for discrete time domain. Fourier transform is represented using $\mathcal{F}$ and inverse Fourier transform using $\mathcal{F}^{-1}$. Fourier transform of a signal denoted by a lower case letter is represented by its upper case letter and in terms of the frequency variable f. i.e. $\mathcal{F}(x(t))(f) = X(f)$ for all $f = 0, \ldots, N-1$.

We define the l^2 norm of a signal x by $||x(t)||_{l^2} := \left(\sum_{i=0}^{N-1} |x(i)|^2 \right)^{\frac{1}{2}}$.

We use $\tilde{(.)}$ to represent an estimate of $(.)$ both in temporal and frequency domains. For any $z \in l^2(\mathbb{Z}_N)$, we define the root mean square $RMS(z) = \frac{||z||_{l^2}}{\sqrt{N}}$.

For computational purposes, we take N to be an integer power of 2 (usually 1024) so that operations like Fast Fourier Transform or Fast Wavelet Transform can be computed quickly.

Definition 1. $l^2(\mathbb{Z}_N) = \{f : \mathbb{Z}_N \to \mathbb{C} | \sum_{i=0}^{N-1} |f(i)|^2 < \infty\}$

where $\mathbb{Z}_N$ is the additive group of integers modulo N.

Definition 2. *A linear filter is a linear transformation from $l^2(\mathbb{Z}_N) \to l^2(\mathbb{Z}_N)$.*

Definition 3. *A time-invariant linear filter T is a translation-invariant linear transformation such that*

$$T \circ \tau_k = \tau_k \circ T$$

for all k where $\tau_k(z)(t) = z(t-k)$ where $z \in l^2(\mathbb{Z}_N)$

Theorem 1. *T is a time-invariant linear filter if and only if there exists an impulse response function $h \in l^2(\mathbb{Z}_N)$ associated with T such that*

$$T(z) = h * z$$

for all $z \in l^2(\mathbb{Z}_N)$.

Proof. See Theorem 2.19 of [5].

1.2 The Problem

The deconvolution problem is to find an estimate $\tilde{x}(t)$ of the desired signal $x(t)$ from the observed signal $y(t)$ when they are related by

$$y(t) = x(t) * h(t) + n(t) \tag{1}$$

where $h(t)$ is the impulse response of the filter and $n(t)$ is noise added to the filter, usually, a white noise with standard deviation σ i.e. $\mathbb{E}|n|^2 = \sigma^2$.

In the Fourier domain, Equation (1) can be written as

$$Y(f) = X(f)H(f) + N(f). \tag{2}$$

1.3 Sample signal: y

To compute relative error, we start with a known signal x, convolve it with an impulse response h and then add a white noise with $\sigma = 5$ to get the test signal y, as shown in Figure 1.

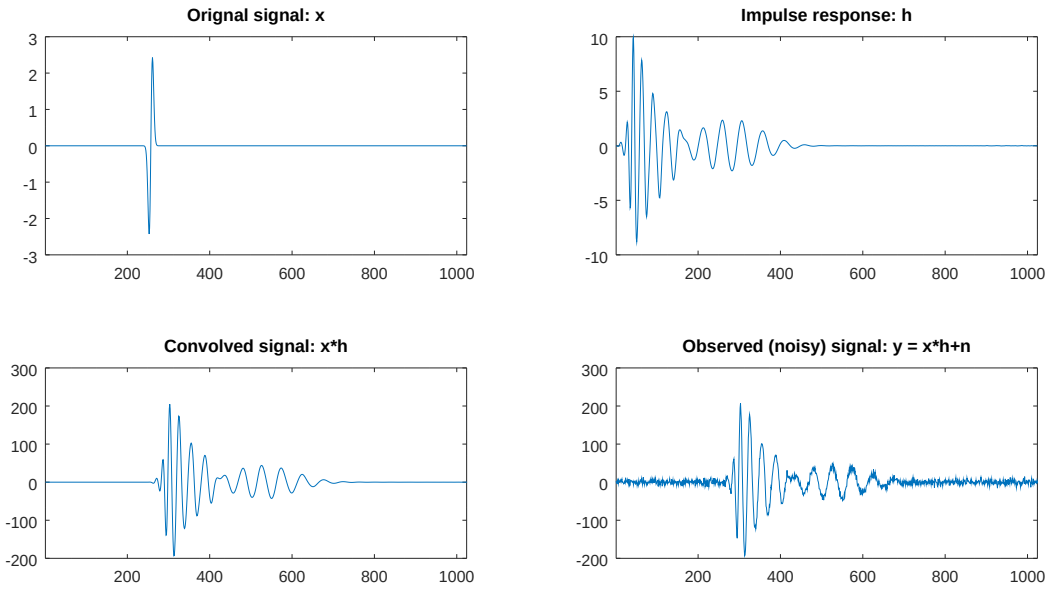


Figure 1: Artificially constructed signal y with noise standard deviation $\sigma = 5$

We choose the width of the spike in x narrow enough so that the frequency support of x contains the frequency support of h. See Figure 2.

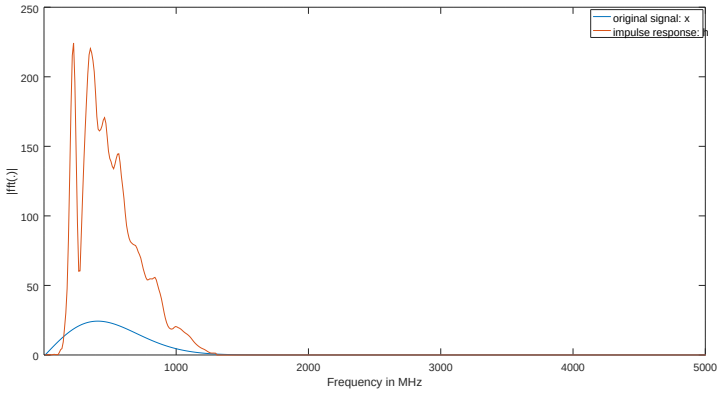
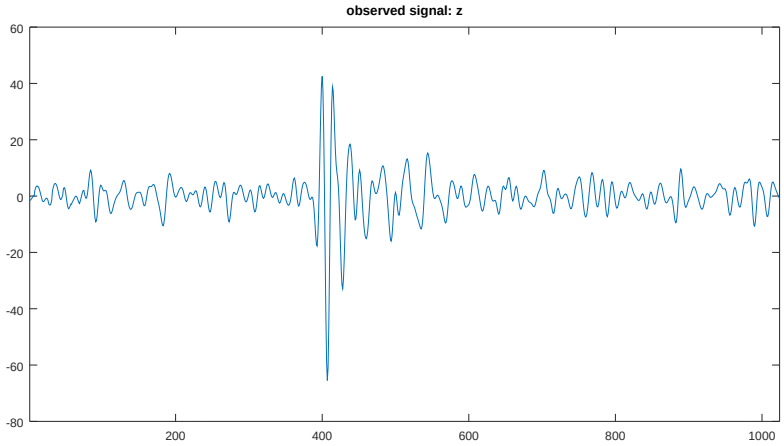


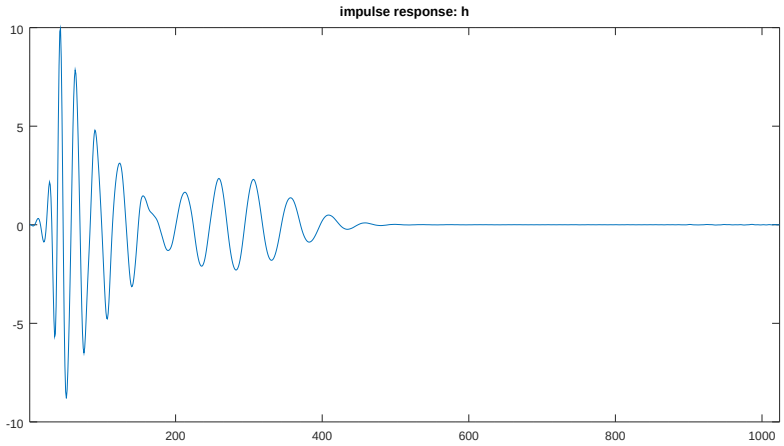
Figure 2: Original signal x and impulse response h in frequency domain. x is chosen such that most of their frequency supports intersect.

1.4 Sample ANITA signals: z

Shown in Figure 3a, we take z to be one of the WAIS pulses observed by ANITA. To construct the signal z, signals from 15 antennas were chosen where the intensity was maximum. The signals were then shifted in time domain to fix the delays and were summed together. Finally, the sum of the signals is upsampled using a sinc interpolation to increase resolution.
The impulse response function h (Figure 3b) is also generated by averaging the impulse responses from all the antennas of ANITA.



(a) Observed signal



(b) Impulse response

Figure 3: Signal z: WAIS event HpolC21203958: $wpx(4,:)(320:320+1023)$ and impulse response h

1.5 Computing signal to noise ratio (SNR)

We use the following method to compute the SNR of a time-localized signal $z(t)$ defined on time interval I

- Remove zero-padding to avoid including silent parts of signal
- Let $M = z(a) = \max(z)$ and $m = z(b) = \min(z)$. Define peak region $p = (b - a)$.
- Compute peak-to-peak distance using $M - m$.
- Remove a neighborhood of length np around the peak region and define the remaining part of the signal by $z_{out} = z|_{I \setminus [a-np, b+np]}$.
  (We have taken $n = 20$ here)
- Compute RMS of z_{out} where $RMS(z_{out}) = \left(\frac{1}{|I|+np} \sum_{t \in I \setminus [a-np, b+np]} |z(t)|^2 \right)^{\frac{1}{2}}$.
- Define $SNR(z) = \frac{M-m}{2RMS(z_{out})}$

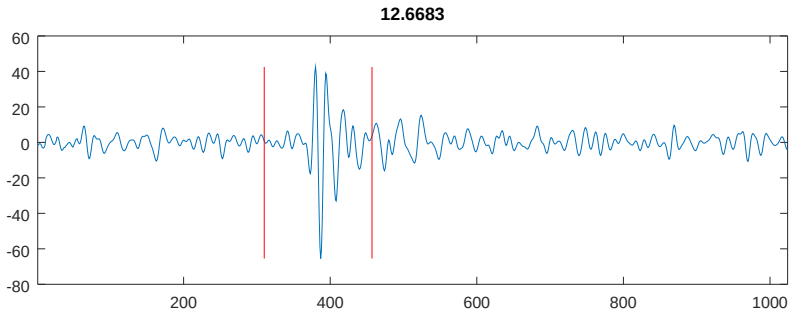


Figure 4: Computed SNR. The region between the red lines are removed while computing the RMS of z_{out}

Definition 4. *Let $\tilde{x}$ be an estimate of $x \in l^2(\mathbb{Z}_N)$. Then, the relative l^2-error is defined by*

$$rerr = \frac{||x - \tilde{x}||_{l^2(\mathbb{Z}_N)}}{||x||_{l^2(\mathbb{Z}_N)}}$$

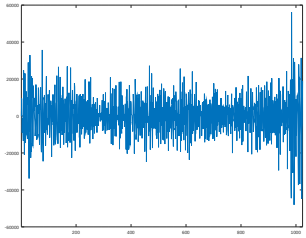
2 Fourier based deconvolution methods

2.1 Naive Deconvolution

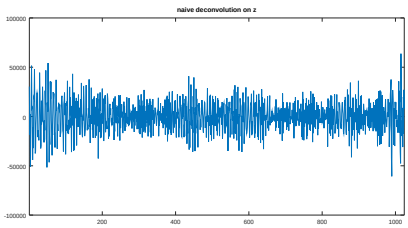
To retrieve X from Y, the most natural step is to divide by H on both sides of (2) to get an estimate $\tilde{X}$ of X like this

$$\tilde{X}(f) := \begin{cases} X(f) + \frac{N(f)}{H(f)} & \text{if } |H(f)| > 0 \\ 0 & \text{otherwise} \end{cases}$$

Figure 5a and 5b shows the result of a naive deconvolution on y.
Issue: When H is small, the noise is significantly amplified in the Fourier domain. This is evident in the figure.



(a) Naive deconvolution on y



(b) Naive deconvolution on z

Figure 5: Naive deconvolution of noisy signals y and z is ineffective

2.2 Fourier shrinkage

To avoid amplifying noise via division, introduce Fourier based regularized deconvolution (FoRD) with an attenuation parameter $\lambda(f) \leq 1$ that is multiplied to $\tilde{X}(f)$.

Then define the estimate of X by $\tilde{X}_\lambda$, where

$$\tilde{X}_\lambda(f) = \tilde{X}(f)\lambda(f). \tag{3}$$

Then,

$$\begin{aligned} \tilde{X}_\lambda(f) &= \left(X(f) + \frac{N(f)}{H(f)} \right) \lambda(f) \\ &= X(f)\lambda(f) + \frac{N(f)\lambda(f)}{H(f)} \\ &= X_\lambda(f) + \frac{N_\lambda(f)}{H(f)} \end{aligned}$$

where $X_\lambda(f) = X(f)\lambda(f)$ and $N_\lambda(f) = N(f)\lambda(f)$.
Thus, the estimated signal $\tilde{x}$ has two parts, which are the retained (attenuated) signal $\mathcal{F}^{-1}(X_\lambda)$ and the leaked noise $\mathcal{F}^{-1}(\frac{N_\lambda}{H})$.

2.3 Examples of $\lambda(f)$

Based on the choice of λ, we have the following Fourier-based deconvolution methods

Definition 5. *Naive deconvolution* $\lambda(f) = 1$

Definition 6. *Allpass deconvolution (from AnitaBuildTools)* $\lambda(f) = |H(f)|$

Definition 7. *Tikhonov deconvolution* $\lambda(f) = \frac{|H(f)|^2}{|H(f)|^2+\tau}$, τ =*constant*

Definition 8. *Wiener deconvolution* $\lambda(f) = \frac{|H(f)|^2}{|H(f)|^2+\frac{N\sigma^2}{|X(f)|^2}}$

Remark 1. *In order to use Wiener deconvolution, we need to supply* $X(f)$ *to obtain the optimal result. However, for practical purposes, since* $|X(f)|^2$ *is unknown, we use* $|Y(f)|^2$ *as a guess. ([6]) Since* $\mathbb{E}|Y(f)|^2 = \mathbb{E}|X(f)H(f)|^2$*, we divide* $|Y(f)|^2$ *by* $||H||^2_{l^2(\mathbb{Z}_N)}$ *to get a better guess for* $|X(f)|^2$*. Instead of* $N\sigma^2$*, we can use* $|N(f)|^2$ *(a frequency-dependent quantity) where* $n(t)$ *is raw noise data, but it does not provide significant improvement. Figure 6 demonstrates the improvement of the Wiener deconvolution method in the context of theoretically constructed signal* y*.*

2.4 Comparison of Fourier based deconvolution methods

In Figure 7, we compare naive (7a, 7d), allpass (7b, 7e) and Wiener deconvolution (7c, 7f) methods on theoretically constructed signal y and on ANITA data z. The performance of naive deconvolution is poor, as expected. The allpass deconvolution scales the signal peak by a factor of the amplitude of the impulse response, as well as introduces artifacts that were not present in the original signal. The Wiener deconvolution performs well. However, Gibbs phenomena are visible near the leading and the trailing edges of the peaks.

2.5 Error analysis

Using $\tilde{x}_\lambda = \mathcal{F}^{-1}(\tilde{X}_\lambda)$, Equation (2), (3) and Plancheral's theorem,

$$\begin{aligned}
||x - \tilde{x}_\lambda||_{l^2} &\lesssim ||X(1-\lambda)||_{l^2} + \|\frac{N_\lambda}{H}\|_{l^2} \\
&= ||x * \mathcal{F}^{-1}(1-\lambda)||_{l^2} + \|\frac{N_\lambda}{H}\|_{l^2} \\
&\lesssim ||X||_{l^1}||1-\lambda||_{l^2} + \|\frac{N_\lambda}{H}\|_{l^2} \text{ (Young's convolution inequality)} \\
&\lesssim ||X||_{l^2}||1-\lambda||_{l^2} + \|\frac{N\lambda}{H}\|_{l^2} \\
&= ||x||_{l^2}||1-\lambda||_{l^2} + \|\frac{N\lambda}{H}\|_{l^2} \\
&= E_{sig} + E_{noi}
\end{aligned}$$

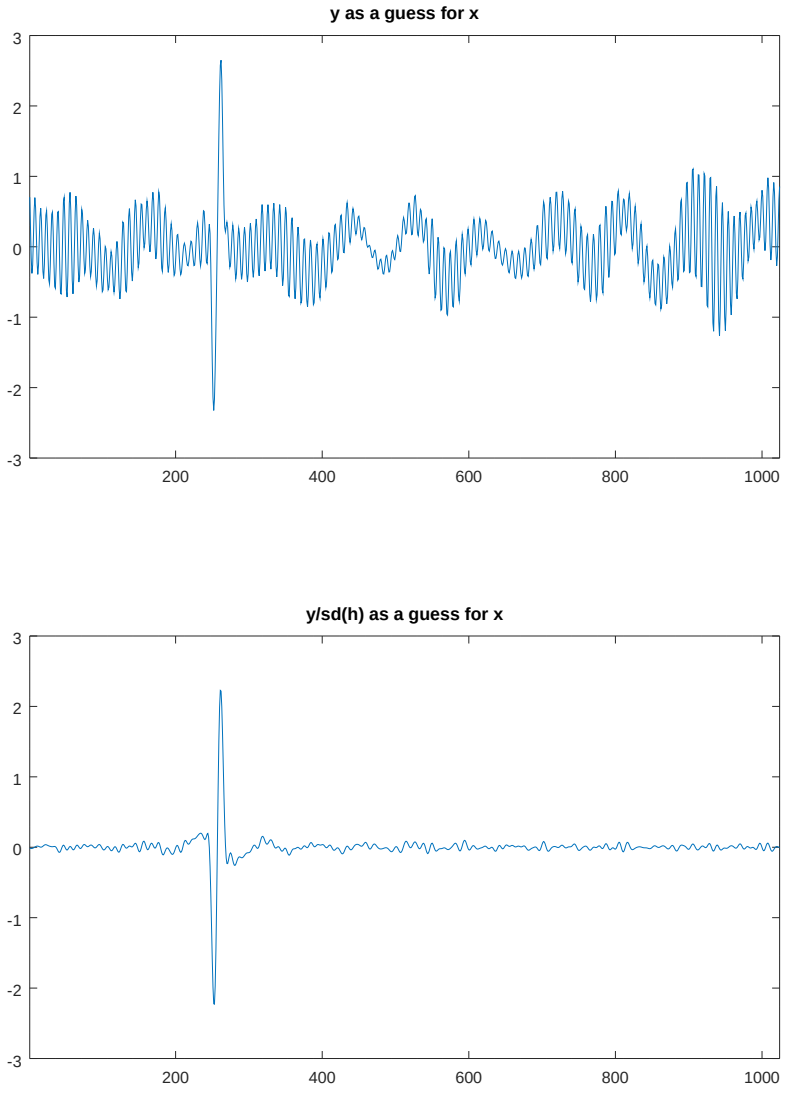
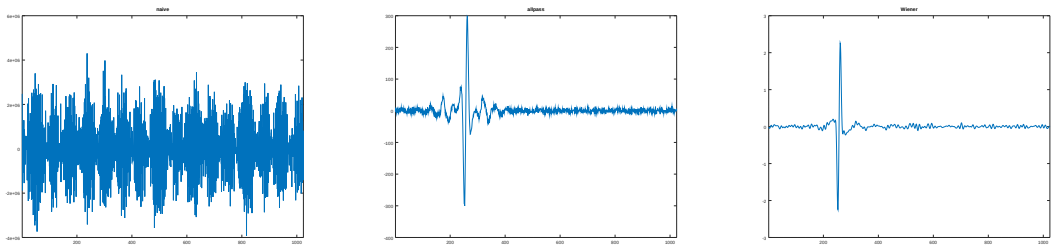


Figure 6: Wiener deconvolution with $|Y|^2$ (top) and $\frac{|Y|^2}{||H||_2^2}$ (bottom) as a guess for $|X|^2$.

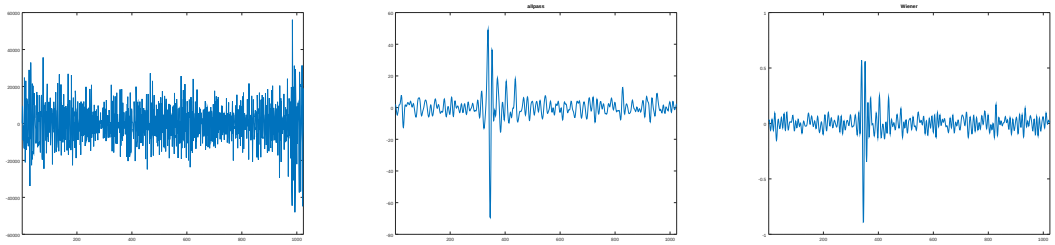
where $E_{sig} = ||x||_{l^2}||l - \lambda||_{l^2}$ and $E_{noi} = ||\frac{N\lambda}{H}||_{l^2}$.

For Naive deconvolution, $\lambda = 1$, hence, $E_{sig} = 0$ and $E_{noi} \lesssim \|NH^{-1}\|_{l^2}$, can be very large depending on H For allpass deconvolution, $\lambda = |H|$. So, $E_{sig} = 0$ only if $|H(f)| = 1$ for all f, which is not necessarily true for an impulse response h.

For any other choice of h, the retained signal part *does not* converge to the original signal. Here, $E_{noi} = ||n||_{l^2}$ For Wiener deconvolution, $\lambda \to 1$ as $\sigma^2 \to 0$, hence $E_{sig} \to 0$.



(a) y - Naive (b) y - Allpass (c) y - Wiener



(d) z - Naive (e) z - Allpass (f) z - Wiener

Figure 7: Comparison of Fourier based deconvolution methods applied to the constructed signal y (top) and Anita signal z (bottom). From left to right: naive, allpass and Wiener deconvolution.

$E_{noi} \leq ||N/H||_{l^2}$ for all n, h. For Tikhnov deconvolution, we assume that the SNR is constant for all frequencies, which is not true for ANITA data.

2.6 Wiener vs allpass deconvolution

We note that for allpass deconvolution, the estimated signal $\tilde{x}$ does not converge to x as $\sigma \to 0$, unless $|H(f)| = 1$ for all f. On the other hand, the Wiener deconvolution is optimal in the following sense:

Theorem 2. *The estimate $\tilde{x}$ obtained via the Wiener deconvolution defined above minimizes the mean square error*

$$\mathbb{E}|\tilde{X}(f) - X(f)|^2$$

among all estimates obtained by applying linear filters to y, for each f.

In Figure 8 and 9 we show the output of allpass and Wiener deconvolution respectively, on 8 of the WAIS pulses observed by ANITA.

We observe that Wiener performs better compared to allpass deconvolution in terms of relative error, whereas in terms of SNR, both methods are comparable.

However, Wiener deconvolution has a few drawbacks. It does not perform well with signals with singularities or "roughness" (high values of derivative) because the energy of the singularity spreads over many Fourier coefficients with significant contributions. Like

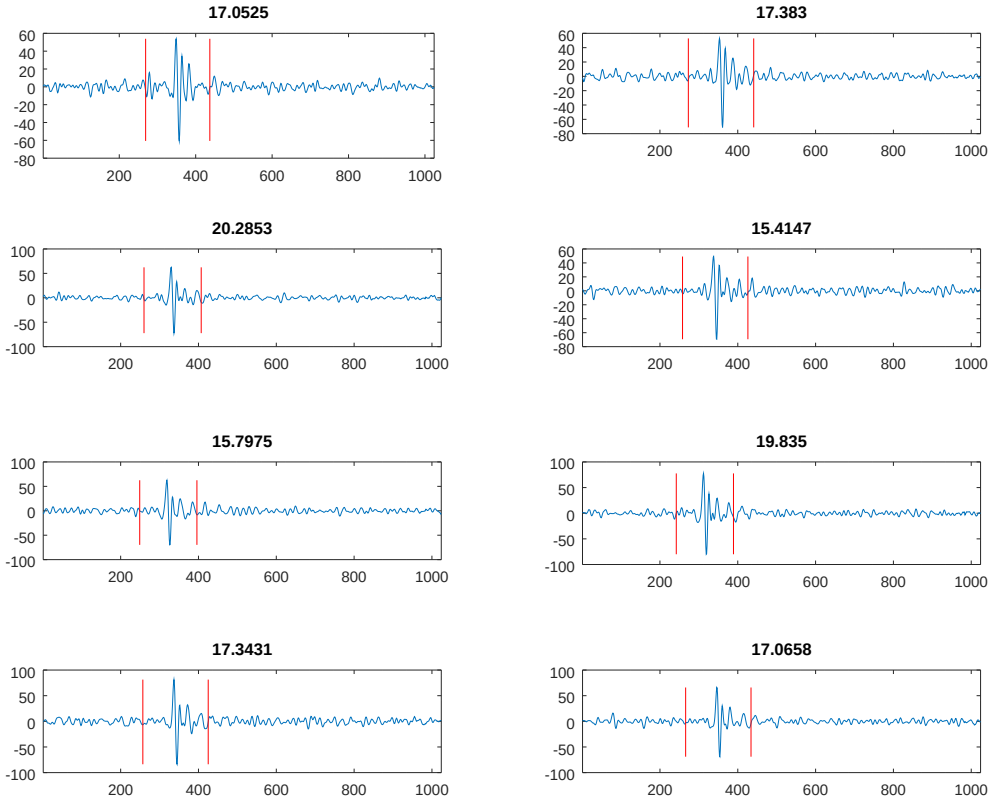


Figure 8: Allpass deconvolution

all Fourier based deconvolution methods, Wiener deconvolution shows ringing round the edges (Gibb's phenomenon) since the basis elements are not temporally localized. Finally, the presence of leaked noise $\mathcal{F}^{-1}(\frac{N_\lambda}{H})$ remains in the output, contributing to the relative error and lowering the SNR.

This motivates us to study the wavelet bases of $l^2(\mathbb{Z}_N)$.

3 Wavelet based deconvolution

We represent the signal in terms of the wavelet basis elements that are localized in the time domain as well as the frequency localized. Thus, rough signals can be represented economically (i.e. using a fewer basis elements). There are wide choices of wavelet bases available to represent the signal. See Figure 10 for a few choices of such basis elements.

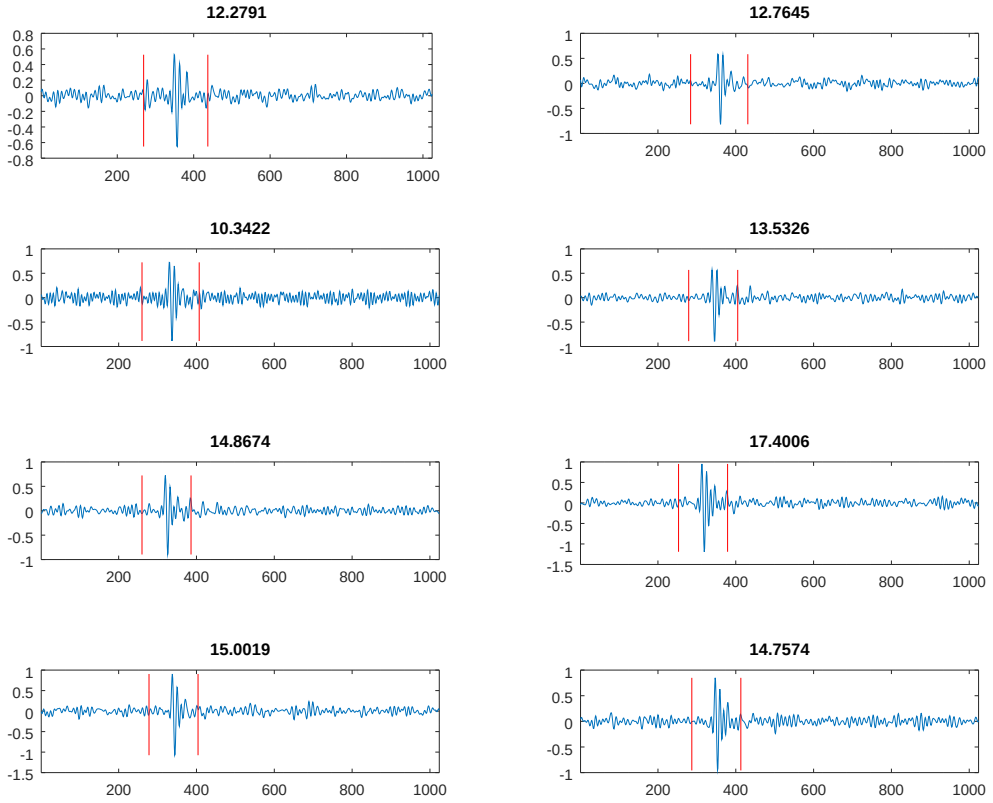


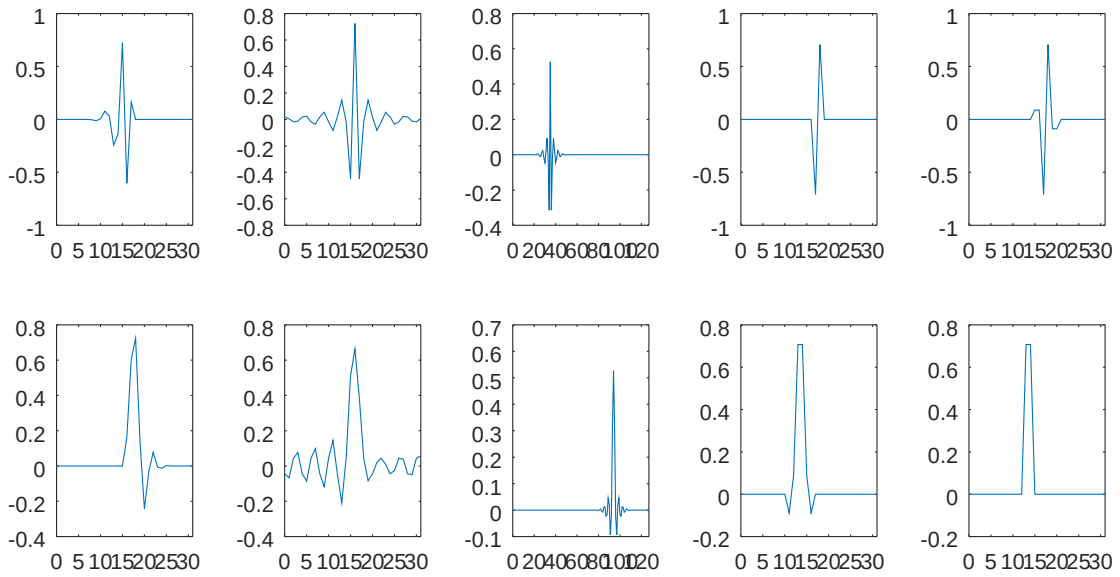
Figure 9: Wiener deconvolution

3.1 Understanding the wavelet basis

For $l^2(\mathbb{Z}_N)$, consider the Euclidean basis which is completely localized in the time domain, i.e., each basis element is supported at one time point. On the other hand, consider the Fourier basis which is completely localized in the frequency domain and thus completely delocalized in the time domain (i.e. spread over the entire time domain) due the uncertainty principle.

Also, note that in the time domain, the elements of the Euclidean basis are formed by starting with one basis element (e.g. $\delta_{1,j}$) and shifting it N times by one unit. On the other hand, the elements of Fourier basis are formed by starting with one basis element (e.g. $e^{-2\pi it/N}$ for each $t = 0, \ldots, N-1$) and then scaling it (i.e., multiplying t by an integer) N times.

Based on the principle that the noise populates any complete orthonormal basis entirely whereas the signal power is localized in the time domain, we look for a complete orthonormal basis whose elements are localized in time domain but are less localized than



(a) D10 (b) Shannon (c) Meyer (d) Bior13-dec (e) Bior13-rec

Figure 10: Wavelet (above) and scaling (below) functions: (from left) Daubechies-10, Shannon, Meyer, Biorthogonal 1.3 (decomposition), Biorthogonal 1.3 (reconstruction)

the Euclidean basis in time domain, so that they are localized in frequency domain as well (by the uncertainty principle). Choosing the Euclidean domain to denoise the signal will modify its frequency information in an undesired way. Similarly, choosing a frequency domain to denoise the signal will fail to capture its time-localized nature.

Unfortunately, we cannot produce a complete orthonormal basis by shifting one function that is less localized than the Euclidean basis elements. However, it is possible to do so by considering two functions and shifting each of them $N/2$ times by 2 units, provided, the two functions φ and ψ follow certain conditions.

More precisely, $\{\tau_{2k}\varphi\}_{k=0}^{N/2-1} \cup \{\tau_{2k}\psi\}_{k=0}^{N/2-1}$ is a complete orthonormal basis of $l^2(\mathbb{Z})$ if and only if the matrix

$$\frac{1}{\sqrt{2}} \begin{bmatrix} \Phi(f) & \Psi(f) \\ \Phi(f+N/2) & \Psi(f+N/2) \end{bmatrix}$$

is unitary for each $f = 0, \ldots, N-1$, where τ_l is a time-domain shift operator by l units. See Theorem 3.8 of [5] for a proof.

Definition 9. *The functions ψ and φ are called the wavelet function and the scaling function respectively. They are also called the generators of the 1st stage wavelet basis for $l^2(\mathbb{Z})$.*

By convention, the wavelet and scaling functions are a high pass and a low pass filters respectively. See Figure 10 for example of such functions.

The vector subspace spanned by $\{\tau_{2k}\varphi\}_{k=0}^{N/2-1}$ is then decomposed further into spans of two families of orthonormal basis elements generated by scalings of φ an ψ. More precisely, $\{\tau_{2^2k}\varphi_2\}_{k=0}^{\frac{N}{2^2}-1} \cup \{\tau_{2^2k}\psi_2\}_{k=0}^{\frac{N}{2^2}-1}$ is an orthonormal basis for the span of $\{\tau_{2k}\varphi\}_{k=0}^{N/2-1}$, where φ_2 and ψ_2 are obtained by scaling φ and ψ respectively. Thus, $\{\tau_{2^2k}\varphi_2\}_{k=0}^{\frac{N}{2^2}-1} \cup \{\tau_{2^2k}\psi_2\}_{k=0}^{\frac{N}{2^2}-1} \cup \{\tau_{2k}\psi\}_{k=0}^{N/2-1}$ is an orthonormal basis of $l^2(\mathbb{Z}_N)$ and is called the 2nd stage wavelet basis.

Continuing in this fashion, the p-th stage wavelet wavelet basis for $l^2(\mathbb{Z}_n)$ is

$$\{\tau_{2^pk}\varphi_p\}_{k=0}^{\frac{N}{2^p}-1} \cup \{\tau_{2^pk}\psi_p\}_{k=0}^{\frac{N}{2^p}-1} \cup \{\tau_{2^{p-1}k}\psi_{p-1}\}_{k=0}^{\frac{N}{2^{p-1}}-1} \cup \cdots \cup \{\tau_{2^2k}\psi_2\}_{k=0}^{N/2^2-1} \cup \{\tau_{2k}\psi_1\}_{k=0}^{N/2-1}$$

where $\psi_1 = \psi$ and ψ_{k+1} is a scaling of ψ_k.

To avoid cumbersome notations, we define $\psi_{j,l} := \tau_{2^jk}\psi_j$ and similarly for $\varphi_{j,l}$. Note that, if $N = 2^n$, we can construct at most the nth stage wavelet basis. Although, in practice, we do not need to go all the way to nth stage.

To conclude, starting from two functions, followed by scaling and shifting, it is possible to construct an orthonormal basis, whose elements are localized both in time and frequency domain. The basis elements as well as the coefficients of each signal with respect to this basis are indexed by two parameters: j and l, related to scaling and shifting (or localization), respectively. Thus, it is convenient to represent the wavelet coefficients in a two dimensional way, where the horizontal dimension represents localization k and the vertical dimension represents scaling j.

See Figure 11 for such a representation of wavelet coefficients of ANITA signal z.

3.2 Wavelet decomposition

Given z and additive noise n, we need to estimate x when

$$z(t) = x(t) + n(t).$$

Given a wavelet function ψ and scaling function φ, a J-th resolution wavelet approximation x^J of x is

$$x^J(t) = \sum_{l=0}^{N_{j_0}-1} s_{j_0,l}\varphi_{j_0,l}(t) + \sum_{j=j_0}^{J}\sum_{l=0}^{N_j-1} w_{j,l}\psi_{j,l}(t)$$

where $N_j = \frac{N}{2^j}$. Here, j represents the scaling (also called the level) and l represents the (temporal) localization at time $2^j l$ of the wavelet bases, j_0 is the coarsest resolution level.

It can be shown that $\lim_{J\to\infty} ||x^J - x||_2 = 0$. The map that takes x to its wavelet coefficients is called the wavelet transform. The wavelet coefficients of a signal of length N can be computed in $O(N)$ complexity using Fast Wavelet Transform (FWT).

Apart from computing the wavelet coefficients via wavelet transform, we can compute them using the inner product with the wavelet basis elements.

$$\begin{aligned} w_{j,l} &= \langle x, \psi_{j,l} \rangle \\ &= \sum_{t=0}^{N-1} x(t)\bar{\psi}_{j,l}(t) \\ &= \frac{1}{N}\sum_{f=0}^{N-1} X(f)\bar{\Psi}_{j,l}(f) \\ &= \frac{1}{N}\sum_{f=0}^{N-1} X(f)\bar{\Psi}_{j,l}(f) \end{aligned}$$

3.3 Choices of wavelet basis to represent given signal

Ideally, the wavelet basis is chosen based on the smoothness requirement of the desired signal. To choose the wavelet basis, two factors are to be fixed: the support of the wavelet basis (the time interval on which the wavelet is nonzero) and the vanishing moment (the function is said to have p-th vanishing moment if up to $(p-1)$th derivative of its Fourier transform at zero is zero. i.e. $\frac{d^j}{df^j}\Psi(0) = 0$ for all $j = 1, 2, \ldots, p-1$). The vanishing moment is related to how fast the function decays at infinity. For example, to approximate the function $\frac{sin(t)}{t^5}$, we need to consider wavelets of vanishing moment less or equal to 4.

The support and vanishing moments are related in the sense that smaller support implies smaller vanishing moment.

3.4 Estimating signals in wavelet domain

From now on, we only write the wavelet coefficients $w_{j,l}$ associated to the basis elements $\psi_{j,l}$, others are similar.

First, we look at the wavelet coefficients of the signal obtained using the Fourier based deconvolution. But these wavelet coefficients have leaked noise introduced by the Fourier based deconvolution. So, this noise needs to be thresholded. More precisely, we first estimate the wavelet coefficients of $w_{j,l}$ of x by $\tilde{w}_{j,l}$ where $w_{j,l}$ is the wavelet coefficient of the signal obtained using a Fourier based deconvolution method. Then, we shrink or threshold $\tilde{w}_{j,l}$ using a wavelet shrinkage parameter $\lambda^w_{j,l}$ where

$$w_{j,l} := \lambda^w_{j,l}\tilde{w}_{j,l}.$$

Finally, we compute the estimate $\tilde{x}^J$ of x from thresholded coefficients using

$$\tilde{x}^J(t) := \sum \tilde{w}_{j,l;\lambda}\psi_{j,l}.$$

3.5 Wavelet shrinkage for denoising

Here are some examples of wavelet shrinkage parameters.

Definition 10. *Oracle thresholding [4]*

$$\lambda^w_{j,l} := \mathbb{I}_{\{|w_{j,l}|>\sigma_j\}}$$

for every l, where σ_j^2 is the noise variance at level j and $\mathbb{I}$ is the indicator function.

This is parameter is impractical since $w_{j,l}$ is unknown. In practice, we take $\tilde{w}_{j,l}$ as a guess.

Definition 11. *Hard thresholding*

$$\lambda^w_{j,l} := \mathbb{I}_{\{|\tilde{w}_{j,l}|>\rho_j\sigma_j\}}$$

where (recall) $\tilde{w}_{j,l} = \langle \tilde{x}, \psi_{j,l}\rangle$ and $\rho_j > 0$ is a scale-dependent thresholding parameter.

Definition 12. *Soft thresholding [3]*

$$w_{j,l;\lambda} := sgn(\tilde{w_{j,l}})(|\tilde{w}_{j,l}| - \rho_j\sigma_j)\mathbb{I}_{\{|w_{j,l}|>\rho_j\sigma_j\}}$$

Definition 13. *Wavelet domain Wiener*

$$\lambda^w_{j,l} := \frac{|w_{j,l}|^2}{|w_{j,l}|^2 + \sigma_j^2}$$

Since $w_{j,l}$ is unknown, do a hard thresholding first with respect to a different wavelet basis to estimate $w_{j,l}$.

In Figure 11, the wavelet coefficients of the signal y are shown for level 1 through 5 and the coarsest level 5. The thresholds are shown using a pair of red lines. After the hard thresholding is applied, the result is shown in Figure 12.

Figure 12 demonstrates the effect of ρ as an extra control over the wavelet shrinkage. High values of ρ produces less noisy results but removes signal power.

3.6 How much to threshold after Fourier shrinkage?

We apply a wavelet shrinkage of $\lambda^w_{j,l} = \mathbb{I}_{\{|w_{j,l}|>\sigma_{j,l,\lambda}\}}$ after performing a Fourier based deconvolution with Fourier shrinkage parameter λ.

Theorem 3. *The variance of the leaked noise at the j-th level is*

$$\sigma^2_{j,l,\lambda} = \sigma^2 \sum_{f=0}^{N-1} \frac{|\Psi_{j,l}(f)|^2}{|H(f)|^2}|\lambda(f)|^2. \tag{4}$$

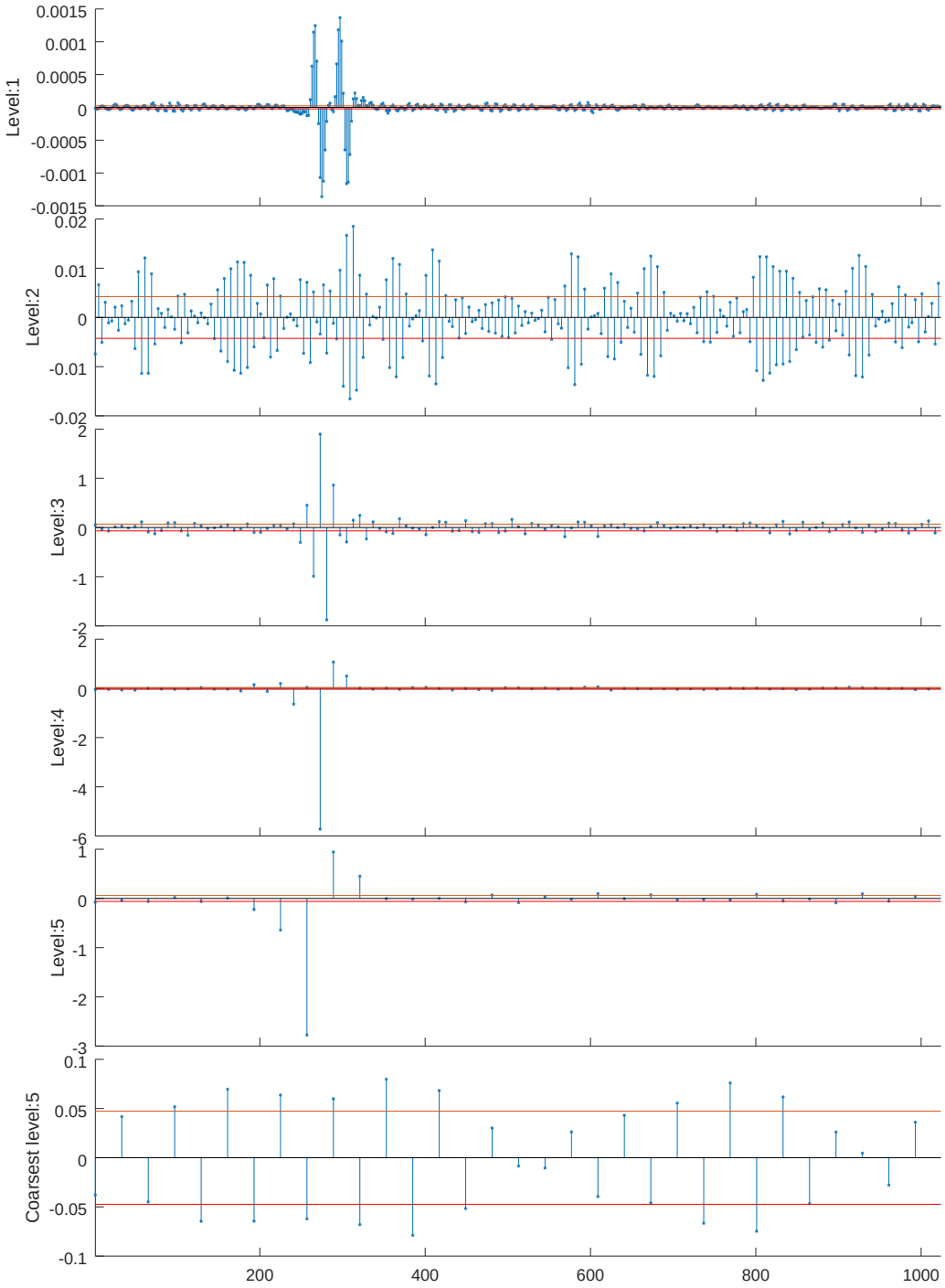


Figure 11: Wavelet coefficients of the 5-th resolution wavelet decomposition of z are shown using a stem plot using blue color. The red lines represent noise standard deviations which will be thresholded later.

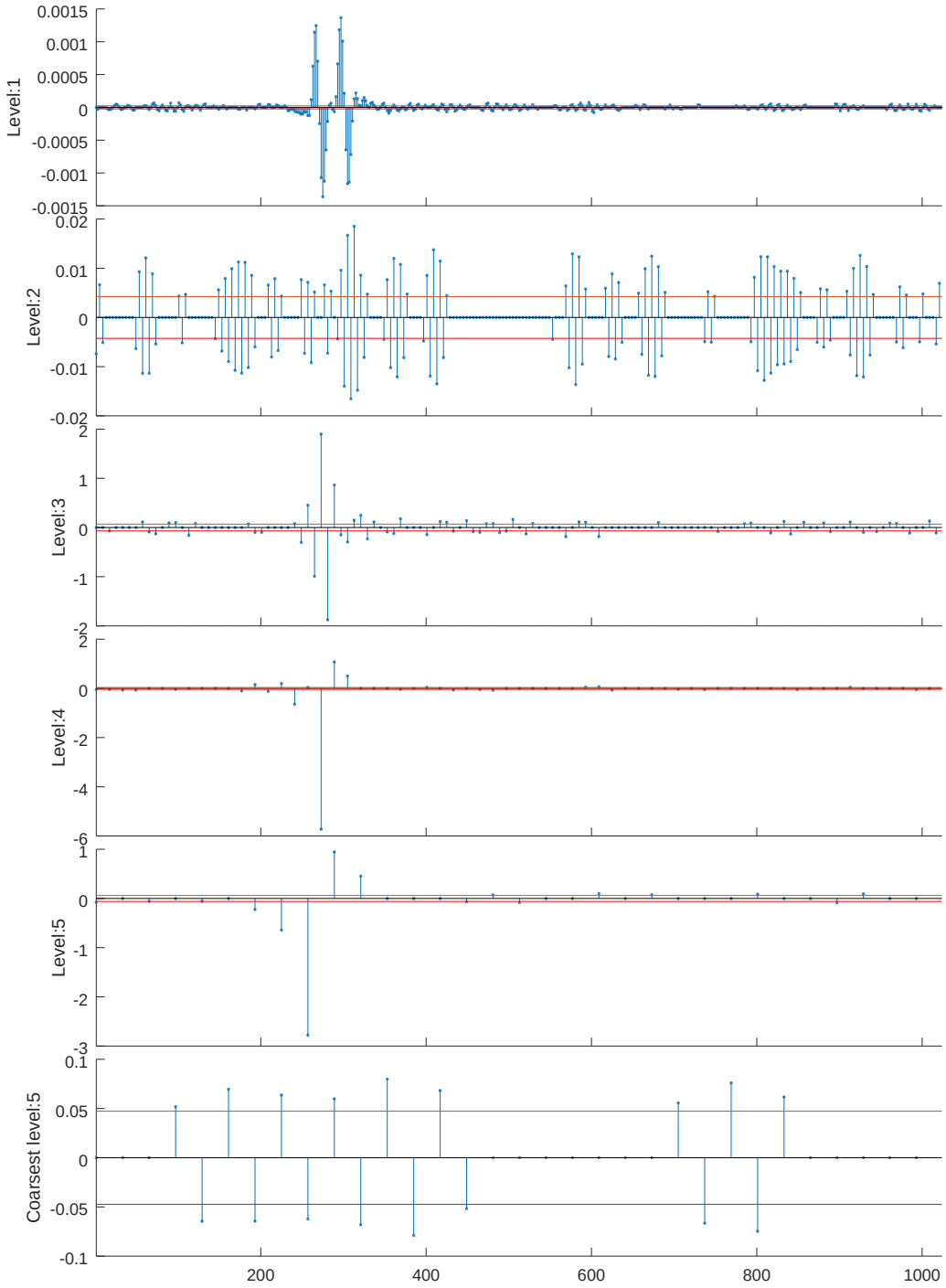


Figure 12: Wavelet coefficients of z are shown using a stem plot using blue color after a hard thresholding has been applied to Figure 11. Any coefficient between the two red lines at each level in Figure 11 is set to zero.

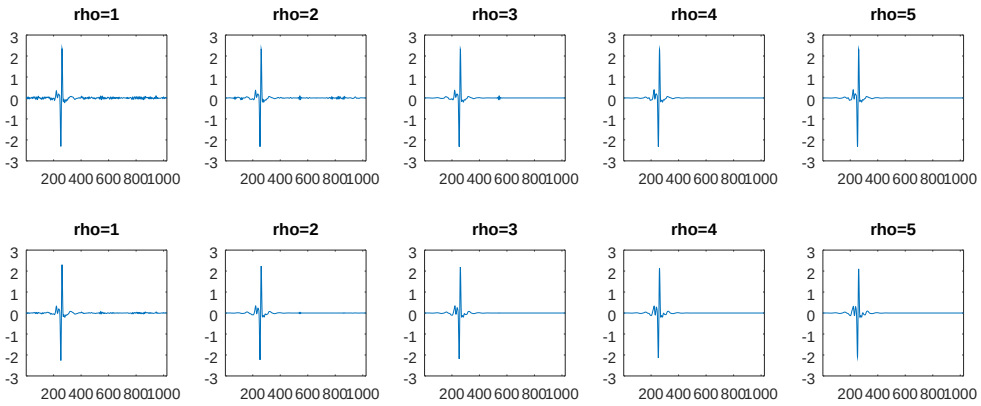


Figure 13: From top to bottom: hard and soft thresholding used in ForWaRD method to deconvolved y. From left to right: $\rho = 1, 2, 3, 4, 5$. In all cases 5-th stage Meyer's wavelet decomposition were used. Soft thresholding is more effective in removing artifacts.

In particular, for Wiener deconvolution, $\lambda = \frac{|H|^2}{|H|^2+\Lambda}$. Thus,

$$\sigma^2_{j,l,\lambda} = \sigma^2 \sum_{f=\frac{N}{2}+1}^{\frac{N}{2}} |\Psi_{j,l}(f)|^2 \left[\frac{|H(f)|}{|H(f)|^2 + \Lambda(f)}\right]^2$$

Remark 2. *When $\Lambda(:= 1/SNR) \to 0$ and the variance of the noise σ remains fixed, we have $\sigma^2_{j,l,\lambda} \to \sigma^2$. Also, when $|H(f)| > 1$ for all f, we have reduced variance of leaked noise $\sigma^2_{j,l,\lambda} < \sigma^2$.*

3.7 ForWaRD: balancing Fourier and wavelet shrinkage

Wavelet-based deconvolution methods have been studied extensively in recent years. See WaveD method [7], wavelet based Galerkin method [2], wavelet frame based deconvolution [1], for example. Here, we consider the Fourier Wavelet based Regularization Deconvolution (ForWaRD) algorithm introduced by [8]. It has the following steps.

1. Do Wiener decovolution with a scaling parameter α_j to estimate j-th level wavelet coefficient of x

$$\tilde{X}(f) = \frac{Y(f)}{H(f)} \frac{|H(f)|^2}{|H(f)|^2 + \frac{N\alpha_j\sigma^2}{|Y(t)|^2}}$$

2. Compute the wavelet coefficient $\tilde{w}_{j,l}$

$$\tilde{w}_{j,l} = \sum_{f=0}^{N-1} \tilde{X}(f)\bar{\Psi}_{j,l}(f)$$

3. Compute thresholded wavelet coefficient $w_{j,l;\lambda}$ using hard, soft or wavelet domain Wiener thresholding and leaked noise variance $\sigma_j(\alpha_j)$ and level-dependent thresholding parameter ρ_j.

4. Reconstruct x using

$$x^J(t) = \sum_{j=j_0}^{J} \sum_{l=0}^{N_j-1} w_{j,l;\lambda} \psi_{j,l}(t)$$

3.8 Optimal choice for α_j

Theorem 4. *[8] For Oracle thresholding, the cost function*

$$\tilde{MSE}_j(\alpha_j) = \sum_{l=0}^{N_j-1} \sum_{j=0}^{N-1} |X(f)|^2 |\Psi_{j,l}|^2 |1 - \lambda(\alpha_j)|^2 + \sum_{l=0}^{N_j-1} \min(|w_{j,l}|^2, \sigma^2_{j,\lambda(\alpha_j)})$$

is minimized if the scaling parameters α_j satisfy

$$\alpha_j = \frac{1}{N_j} \#\{|w_{j,l}| > \sigma_j(\alpha_j)\} \tag{5}$$

Remark 3. *Wiener deconvolution corrupts the signal power associate with the frequency carrying noise. Thus, Equation (5) implies that signals with more economical wavelet representations should require less Fourier shrinkage and vice versa. Thus, a signal with larger number of unthresholded wavelet coefficients imply that that signal required more Fourier shrinking.*

Unfortunately, we do not have any analytical formula for α_j. However, α_j's are independent, hence, they can be computed using bisection method or Newton's method, making the process computationally expensive. Thus, instead of computing the scaling parameters α_j's for all each signal, we choose a set of signals first and plot the distribution of the scaling parameters. Then, we choose a reasonable statistics (for example, median) as an estimate of the scaling parameters for the rest of the signals.

In Figure 14, we show the distribution of $\{\alpha_j\}_{j=1}^6$ of arbitrarily chosen eight WAIS pulses. Although the sample size considered here is too small, this method will give up a reasonable "guess" for α_j to be used in ForWaRD algorithm applied to other WAIS signals.

3.9 Comparison of ForWaRD, Wiener and allpass deconvolutions for y

In Figure 15, we apply 3 deconvolution methods to the simulated signal y and compare the relative error and SNR. We generate noise of various standard deviation to create the simulated data.

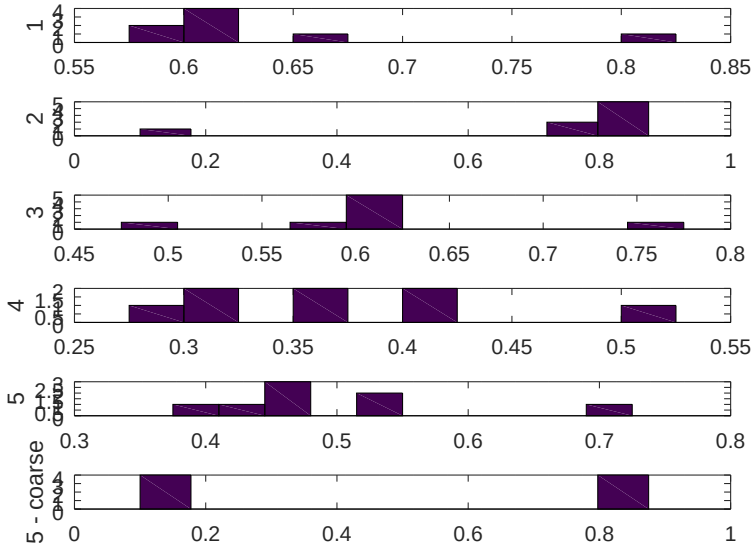


Figure 14: Distribution of the scaling parameters α_j of 8 wavefoms. The level 5 - coarse is dropped so the value does not matter.

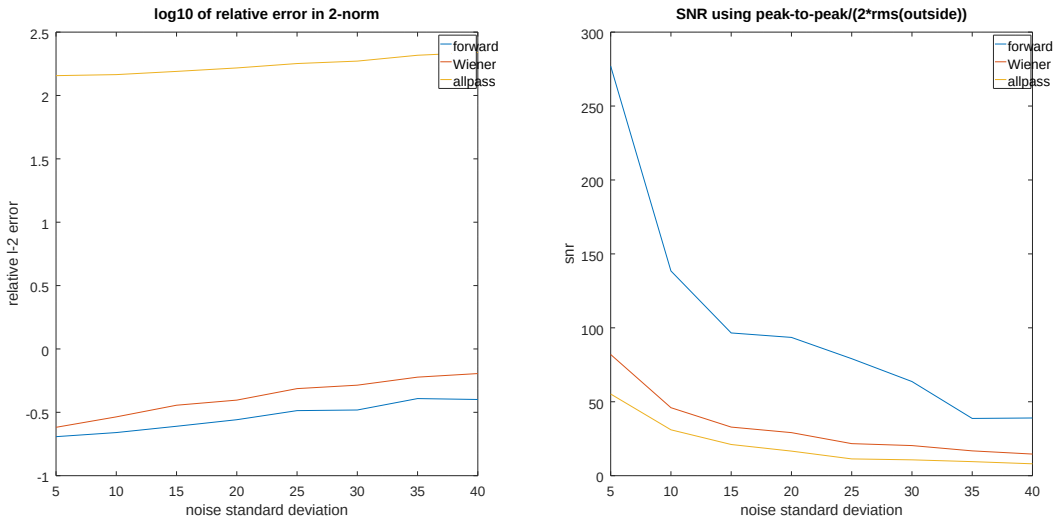
As the noise level increases, the deconvolution methods get less effective (i.e. the relative error increases). However, in all noise levels, ForWaRD method performs better compared to other two deconvolution methods.

3.10 Parameters of ANITA data

- ANITA hardware has an effective passband of 180-1250 MHz.
- The sample rate of the signals is $f_s = 10^{10}$ samples/s. So, the Nyquist frequency is 5×10^9 GHz or 5000 MHz.

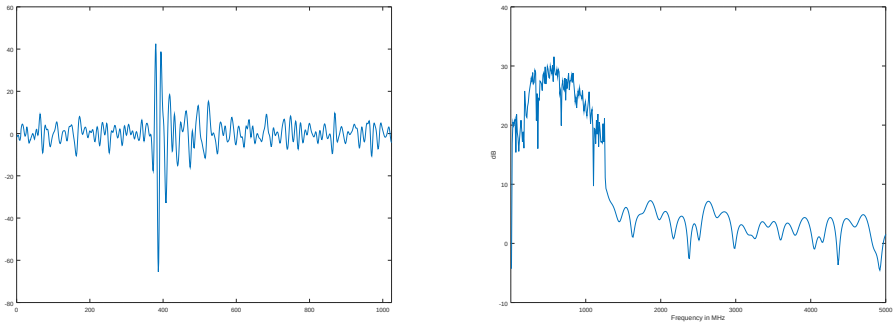
3.11 Deconvolving ANITA signals

To deconvolve ANITA data using ForWaRD algorithm, we use the noise data captured by ANITA to estimate noise power. We start with a guess for α_j and apply the ForWaRD algorithm using the modified Wiener deconvolution and 5-th resolution Meyer's wavelet basis and soft thresholding. We iterate using bisection method to get the optimum scaling parameters α_j. Finally, we apply an aggressive thresholding on 5th level Meyer's wavelet basis to punish frequency ranges 0-200 MHz. Finally, we project the output onto the 2nd level Meyer's wavelet level to punish frequency ranges 1000 - 5000 MHz (approximately). Alternatively, we apply a Plank-taper window (a smooth Fourier cut-off function) to remove unwanted frequency ranges from the output signal.



(a) Relative error in $\log_{10}$ scale (b) Signal-to-noise ratio

Figure 15: Relative error and SNR are compared among ForWaRD, Wiener and allpass deconvolution. For all noise variance, ForWaRD produces lowest relative error and highest SNR. A level-dependent threshold was used for ForWaRD method.



(a) Signal z (b) Power spectrum of z.

Figure 16: A typical ANITA signal of sample length 1024. The power spectrum is shown below in dB.

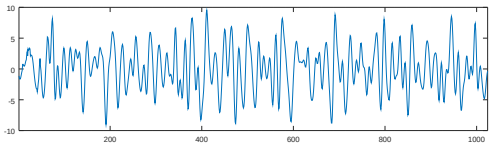


Figure 17: Example of noise data captured by ANITA to estimate noise power.

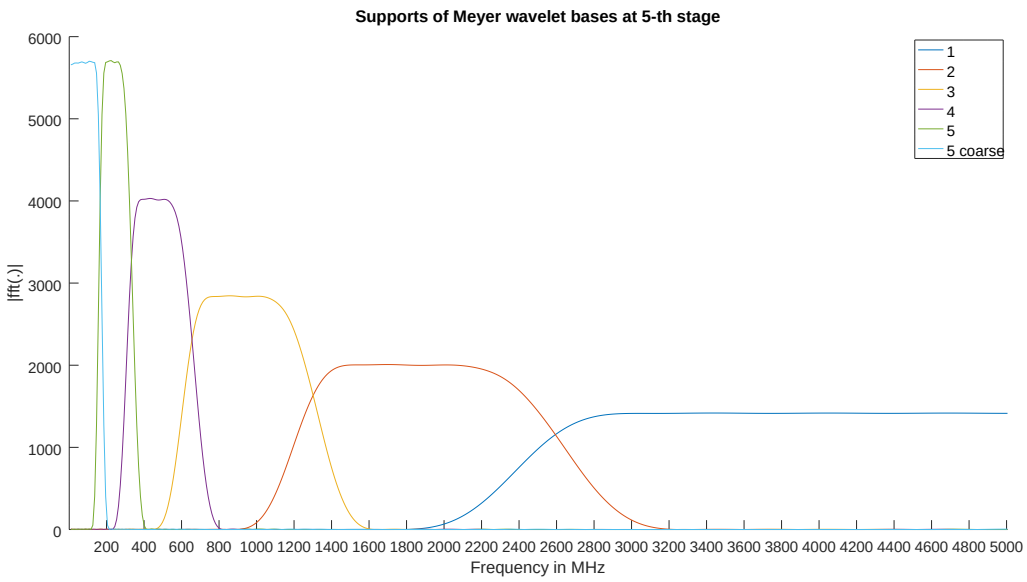


Figure 18: Frequency supports of Meyer's wavelet basis elements of different stages up to 5-th stage. E.g., $P_{-5}(z)$ has a frequency support approximately $[0, 200]MHz$ that can be dropped. Similarly, $P_{-2}(z)$ will have frequency support approximately $[0, 1600]MHz$.

3.12 ForWaRD method applied to ANITA signal z

Figure 20 is the output of the ForWaRD algorithm applied to ANITA signal z. We determine the scaling parameters $\{\alpha_j\}$ using a Newton's method. Here we have used a 5-th resolution wavelet decomposition and Meyer's wavelets.

In Figure 21, we show the observed ANITA signal z in different wavelet levels and the corresponding spectrum.

In Figure 22, we show the wavelet decomposition of the deconvolved signal (Fig 20) in different levels and the corresponding spectrum.

3.13 Effect of ρ_j on ANITA data

The presence of noise in the non-peak region of the output of the ForWaRD algorithm (as seen in Figure 20) suggests that the noise variance was not chosen appropriately. However, looking at the output signal at various wavelet levels, we can adjust the values of ρ_j to get rid of this noise. Figure 23a is an example of sample values of ρ_j that produce a signal with much higher SNR compared to default values of ρ_j (which is $\rho_j = 1$ for all j).

3.14 ForWaRD algorithm applied to eight WAIS pulses

In Figure 24, we apply the ForWaRD algorithm to 8 WAIS pulses and show the output. We use Meyer's wavelets and consider wavelet decomposition till level 5. The values of

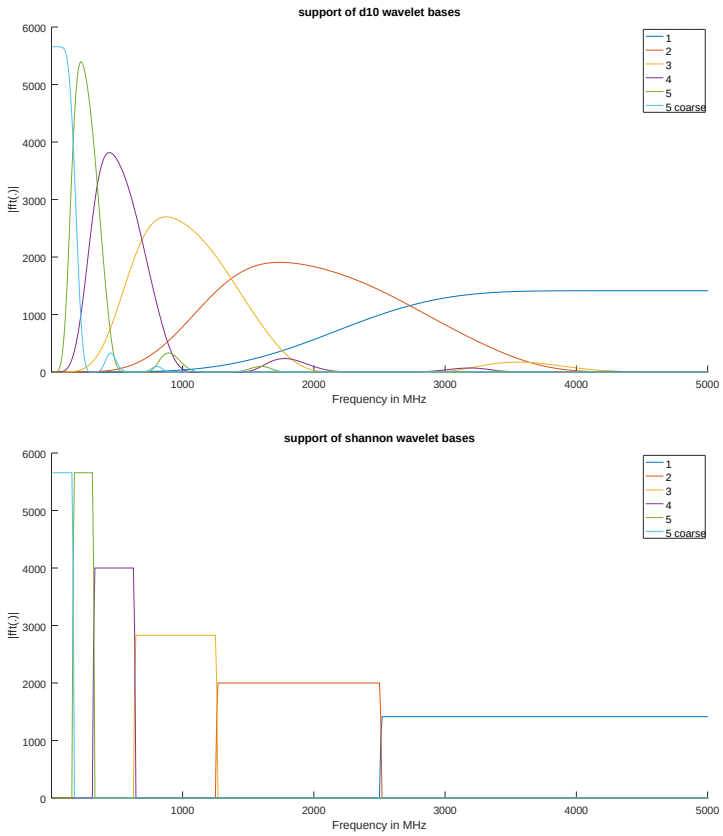


Figure 19: Support of other wavelet bases: D10 and Shannon

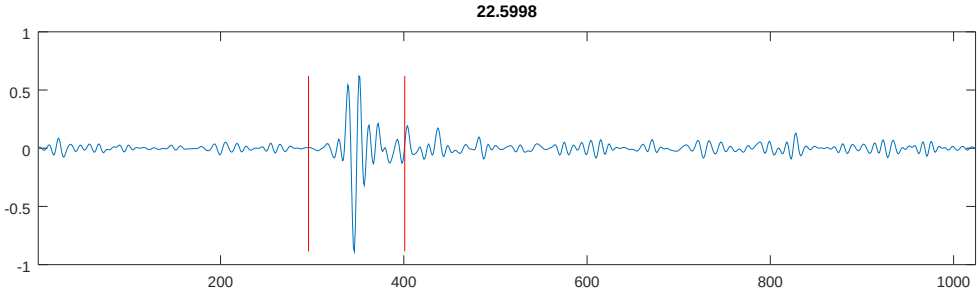


Figure 20: ForWaRD method has been applied to z with $\rho = (1, 1, 1, 1, 1, 10e10)$ (i.e. 5-th coarse level wavelet coefficients were removed) and the output was projected onto 2nd stage wavelet subspace (i.e. 1st and 2nd level coefficients were removed). The SNR is written above. There are some artifacts in the non-peak region.

$\rho_j = 1$ for all j and α_js are computed using a bisection method.
The SNR are printed on top of each signal.

Next, we use the modified values of ρ_j's ($\rho = (1, 1, 4.5, 3, 2, 10e10)$) that we determined

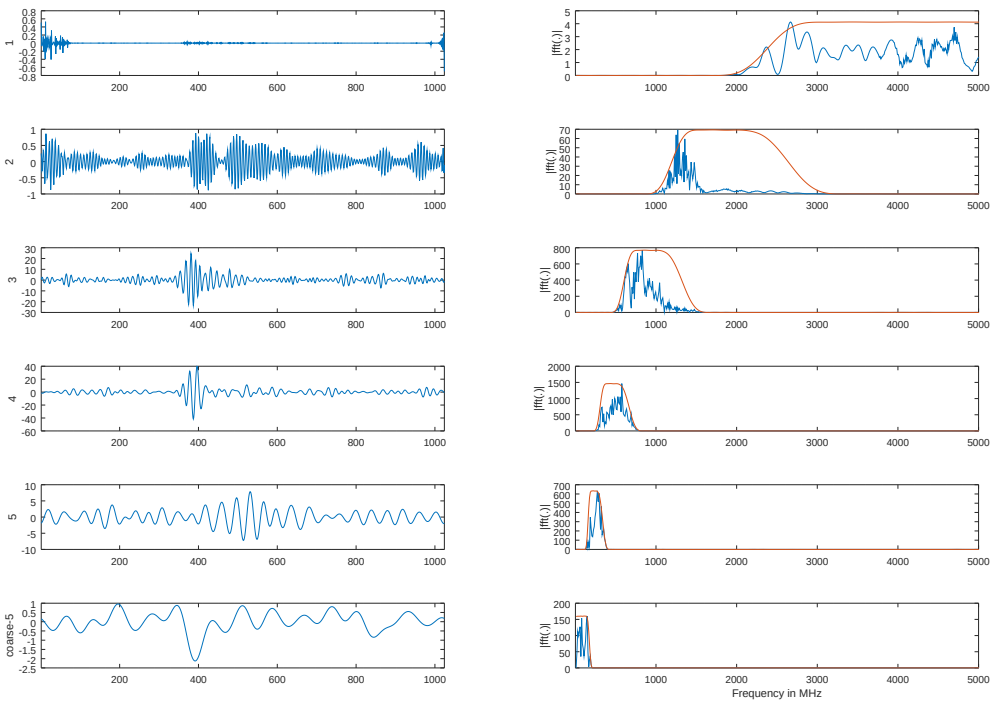


Figure 21: Decomposition of signal z (left) and the powers (right) of the corresponding levels. The frequency ranges of each level is shown using red curves.

from applying ForWaRD algorithm to signal z and apply it to 8 WAIS signal. The outputs are shown in Figure 25.

This choice of ρ seems effective in removing noises in the non-peak regions for all these signals. Further modification can be made to ρ to remove artifacts observed in the 8th signal, for example.

3.15 Other wavelets: Daubechies 10, Shannon

The correct choice of wavelet basis depends on the smoothness requirement of the desired signal. In the case of ANITA data, the smoothness of the signal is unknown.

In Figure 26, we apply the ForWaRD algorithm with Daubechies 10 wavelets instead of Meyer's wavelets, while keeping all the parameters same as in Figure 25, whereas, in Figure 27 we use Shannon's wavelets.

Since any choice of wavelets will generate an orthonormal basis of $l^2(\mathbb{Z}_N)$, the main features of the signal are the same in each case. However, whenever a part of the signal is represented by a very few wavelet basis elements, the shape of the chosen wavelets are visible, causing visual difference.

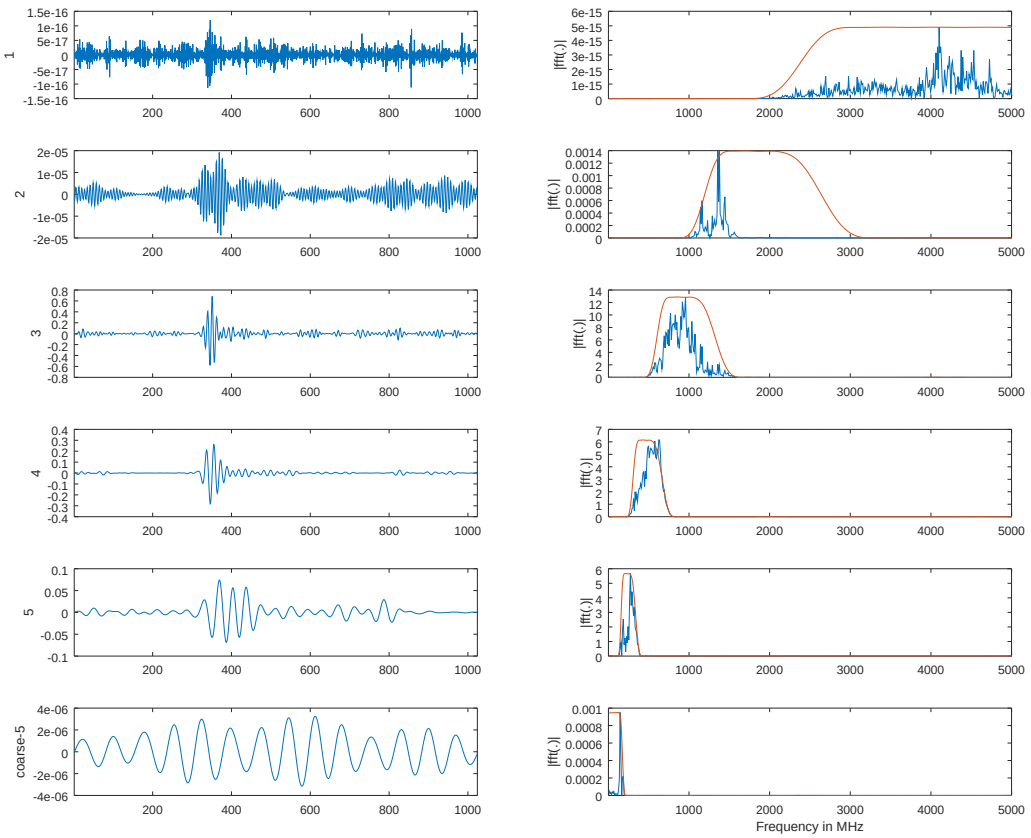
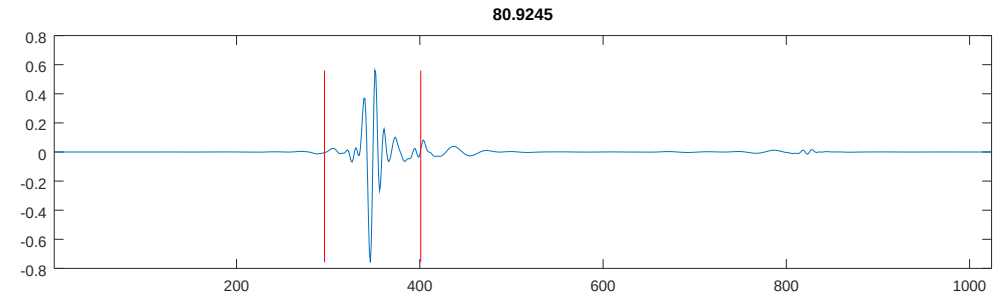


Figure 22: Wavelet coefficients of level 1,2 and 5 are suppressed in the signal. Level 3 has most contribution to the artifacts in the non-peak region.

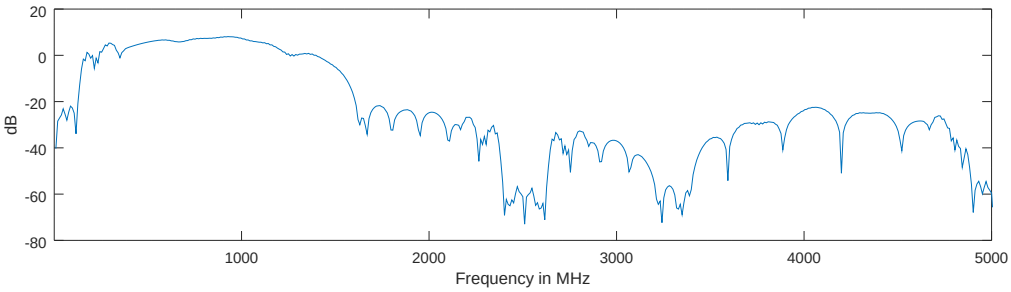
It is unclear at this moment what the right wavelet basis should be. However, assuming that the profile of the ANITA signal are similar to that of the simulated signal y, we observed that out of the three wavelet families considered here, the smallest relative error is achieved for Meyer's wavelets. Thus, we choose Meyer's wavelets to be the most appropriate wavelets for deconvolving ANITA signals. However, further investigation is needed to look for some smoothness information about the ANITA signals.

4 Model for deconvolution with multiple antennas

So far, for the sake of simplicity, we have considered z to be *the* observed signal and supplied the impulse response K to the ForWaRD algorithm to perform deconvolution. However, z is actually the average of 15 signals collected from 15 separate antennas, all of which observed the same event. Each of the antennas have their own impulse response. In



(a) Output of ForWaRD algorithm applied to z with modified ρ



(b) Power spectrum of Figure 23a

Figure 23: ForWaRD method has been applied to z with $\rho = (1, 1, 4.5, 3, 2, 10e10)$ and the output was projected onto 2nd stage wavelet subspace. The output signal (above) and the power spectrum (below) are shown. The SNR is written above.

this section, we apply the ForWaRD algorithm to each of those 15 antennas to deconvolve each signal and compute the average of the outputs.

First, we fix the notations.

Let x_i be the signal arriving at the i-th antenna, where $i = 1, \ldots, M$. (For ANITA signals, $M = 15$).

Let y_i be signal observed by the i-th antenna and h_i be the impulse response of the i-th antenna.

Our goal is to estimate x, the average of x_i's. i.e.

$$x = \frac{1}{M} \sum_{i=i}^{M} x_i.$$

- **Case 1:** (average-deconvolve-denoise) This the case we have considered in the previous sections. The aim was to estimate $\frac{1}{M} \sum_{i=1}^{M} x_i$ where

$$\frac{1}{M} \sum_{i=1}^{M} y_i = \left(\frac{1}{M} \sum_{i=1}^{M} x_i \right) * \left(\frac{1}{M} \sum_{i=1}^{M} h_i \right) + \frac{1}{M} \sum_{i=1}^{M} n_i$$

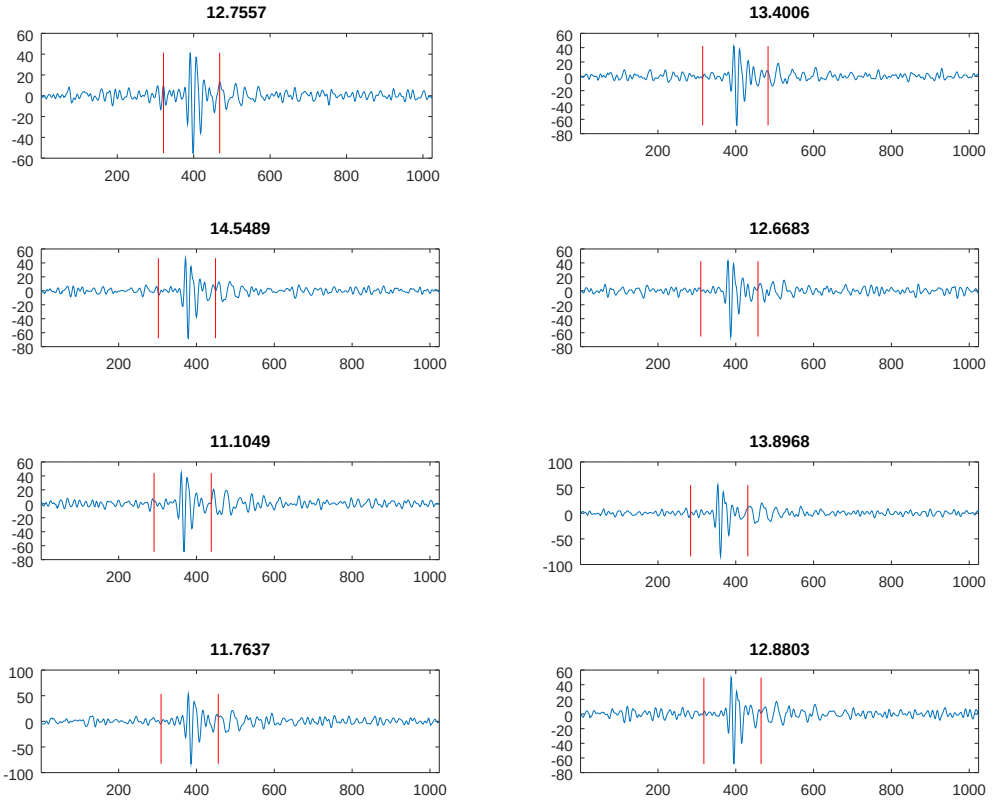


Figure 24: We use 8 WAIS pulses observed by ANITA to test the parameters

- **Case 2:** (deconvolve-denoise-average) This is the case we will be considered in this section. The aim is to estimate $\frac{1}{M}\sum_{i=1}^{M} x_i$ where

$$y_i = x_i * h_i + n_i \text{ for each } i = 1, \ldots, M$$

On the Fourier domain, the estimate $\tilde{X}$ can be represented by

$$\frac{\sum_{f=1}^{N}(Y_i(f) - N_i(f))}{\sum_{f=1}^{N} H_i(f)} \text{ for Case 1}$$

$$\sum_{f=1}^{N} \frac{Y_i(f) - N_i(f)}{H_i(f)} \text{ for Case 2}$$

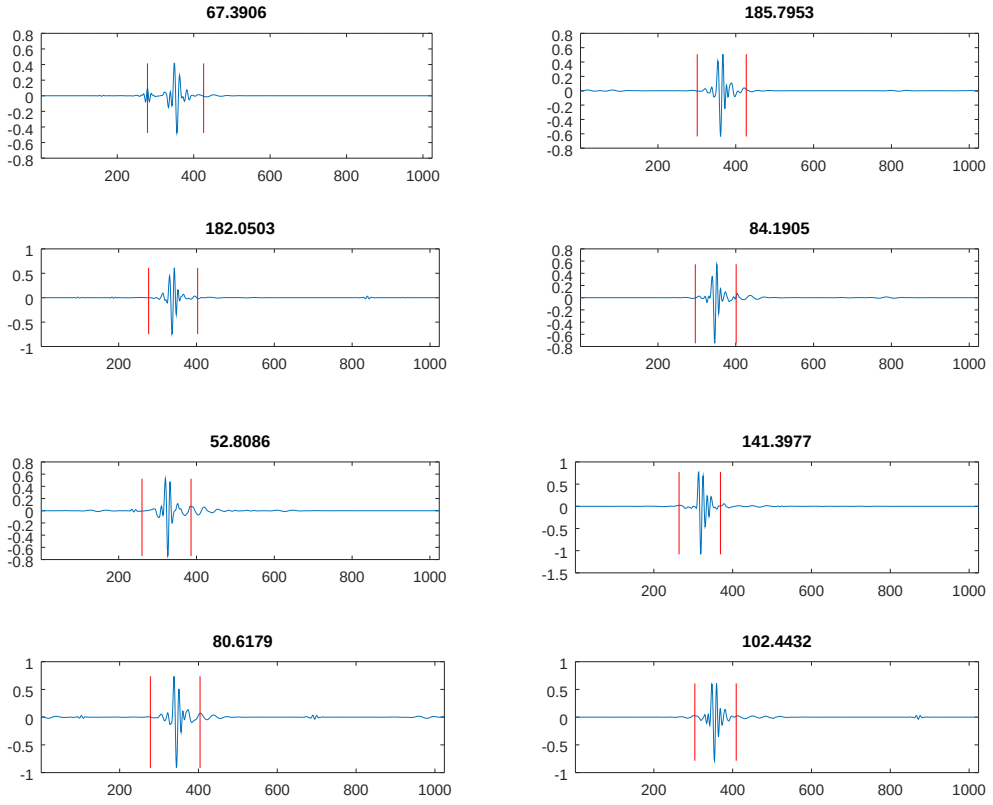


Figure 25: ForWaRD method applied to the WAIS signals

4.1 ForWaRD applied to multiple antennas (i.e. Case 2)

In Figure 28, the ForWaRD algorithm is applied to each of the 15 observed signals y_i and then the average is taken over the outputs of the ForWaRD algorithms. The impulse response and the noise standard deviation recorded by each antenna is supplied to the algorithm. Note that the artifacts in the off-peak region do not sum as destructively as in Case 1. The accumulation of spikes at the beginning and the end is due to zero-padding in the observed signal.

A heuristic explanation of the output of Case 2 (Figure 28) having less SNR compared to the output of Case 1 (Figure 23) is that the noise is considered to be independent at each time point, thus they add up destructively in Case 1 before we apply the deconvolution algorithm. But, in Case 2, the deconvolution algorithm removes the independent noise present in each signal. So, the averaging effect is not destructive enough afterwards, causing a lower SNR value.

More precisely, if $\{n_i\}_{i=1}^{M}$ are i.i.d. random variables with standard deviation σ, $\sum_{i=1}^{M} n_i$ is a random variable with standard deviation $\sqrt{M}\sigma$. Thus the noise standard deviation of

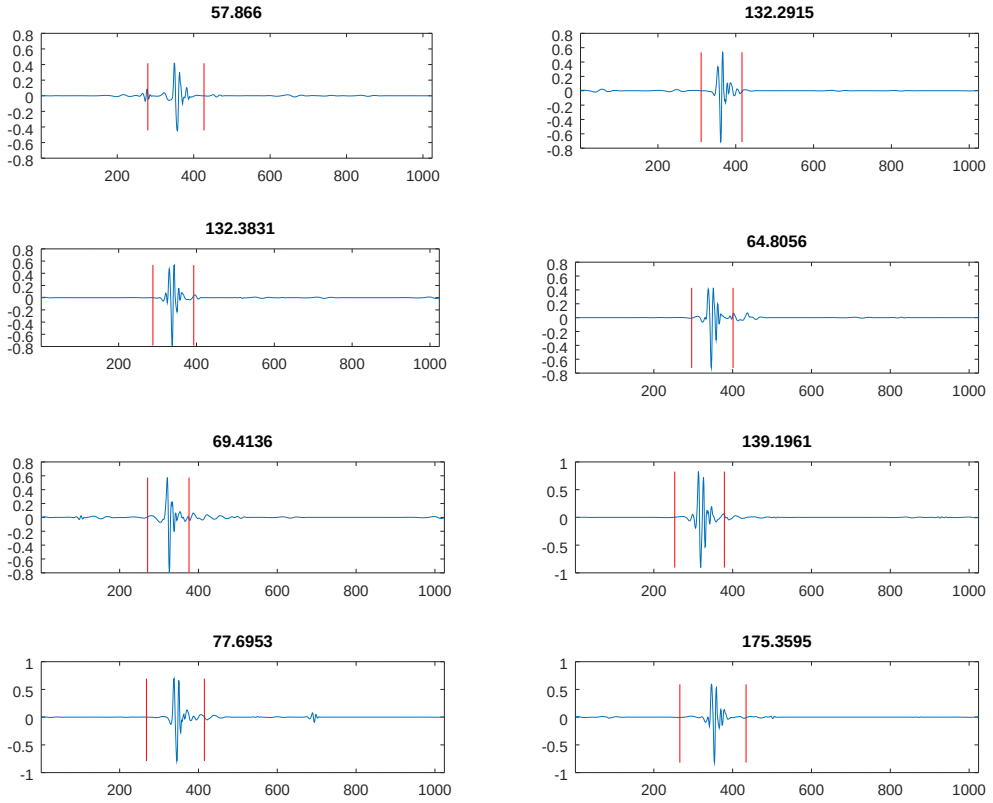


Figure 26: ForWaRD with D10 wavelet basis, same parameters

$\sum_{i=1}^{M} y_i$ is amplified by a factor of $\sqrt{M}$, whereas, the peak of the signal increased by a factor on M. So, summing the noisy observed signals increases the SNR before performing a deconvolution. (See 23)

4.2 Comparison of Case 1 and Case 2: different deconvolution methods

In Figure 29, 30 and 31 we compare the outputs of Case 1 and Case 2, when a allpass, wiener and ForWaRD deconvolution is applied.

We observe that the difference between the outputs of Case 1 and Case 2 in terms of SNR is negligible for Fourier bases deconvolution methods since they are linear with respect to the observed signal, whereas, the wavelet based ForWaRD algorithm, being nonlinear in nature, produces significantly different results.

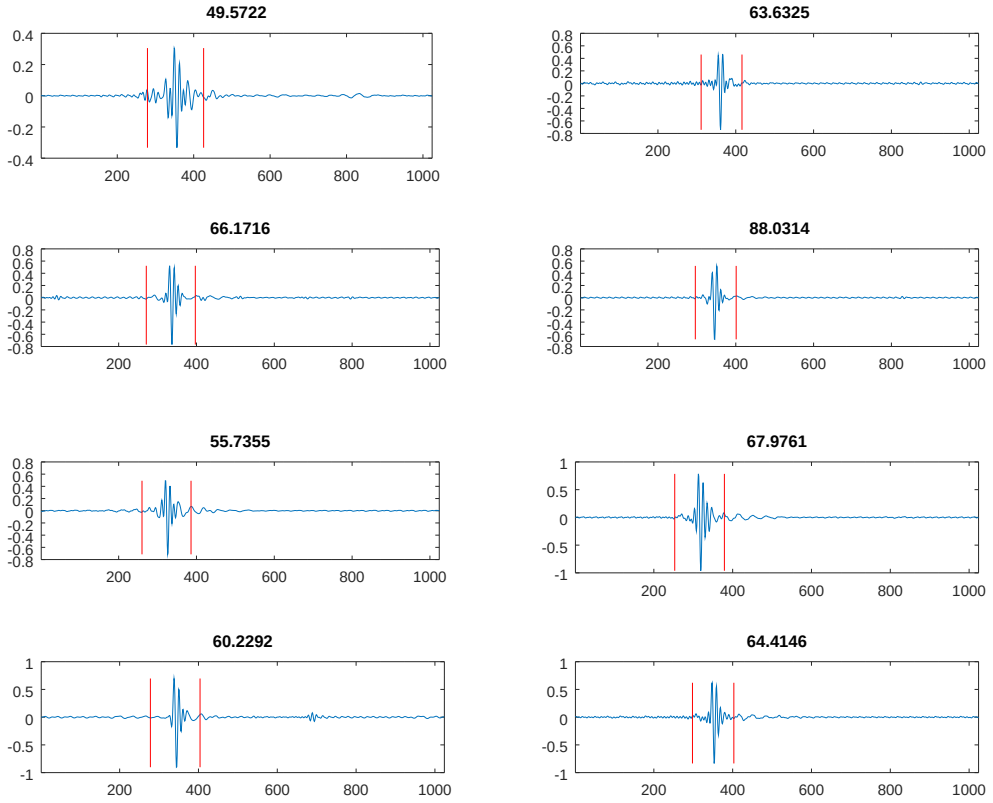
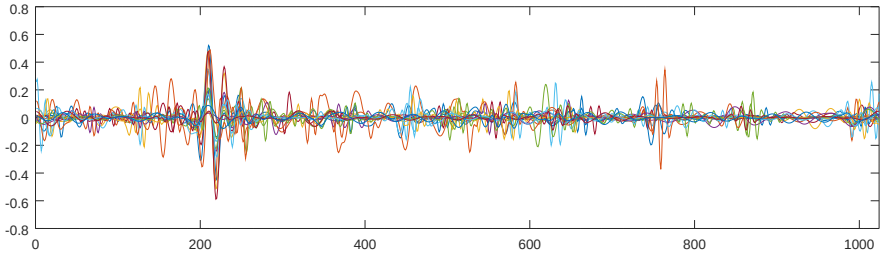


Figure 27: ForWaRD method with Shanon wavelet basis, same parameters

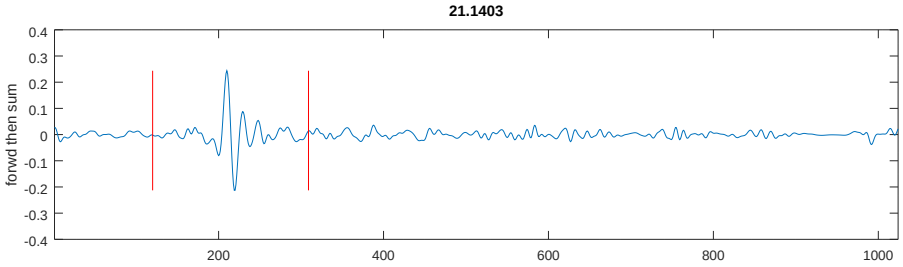
5 Summary

5.1 Conclusion

- Allpass deconvolution does not preserve the l^2 norm of original signal.
- Wiener deconvolution converges to the original signal when noise-to-signal ratio approaches zero.
- Both Wiener deconvolution and ForWaRD method produce higher signal-to-noise ratio and lower relative error than allpass deconvolution for theoretically constructed signals.
- Level-dependent thresholding removes artifacts and improves relative error (for simulated signal) and SNR (for both simulated signal and ANITA signal).
- For ANITA data, theoretical deconvolved signal is not available hence cannot compute the relative error. Thus, we rely on results from simulated signal y.

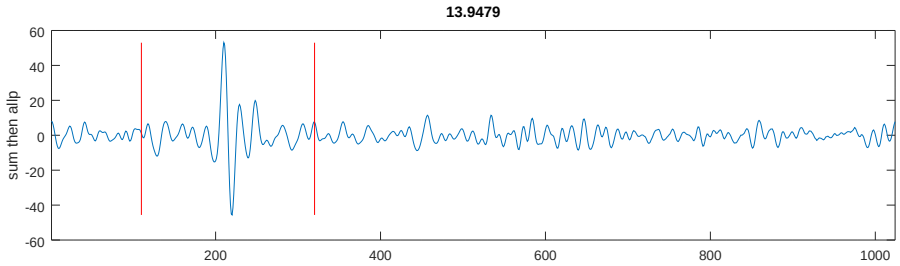


(a) Superposition of individually deconvolved signals using ForWaRD algorithm

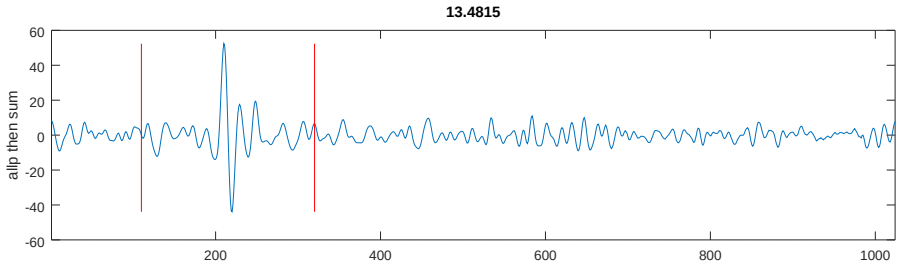


(b) Average of 15 deconvolved signals shown in Figure 28a

Figure 28: Output of multi-antenna model: ForWaRD-deconvolved individual antenna signals and their average



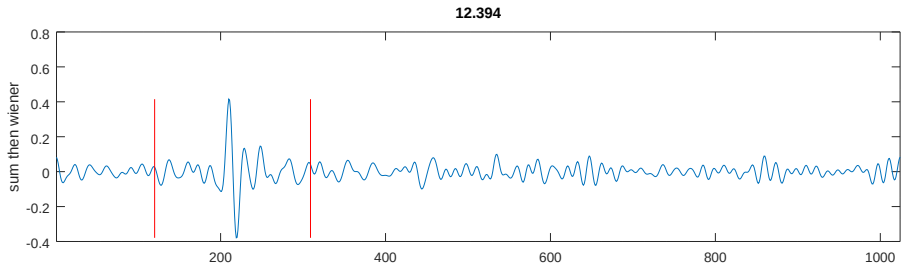
(a) Allpass - Case 1



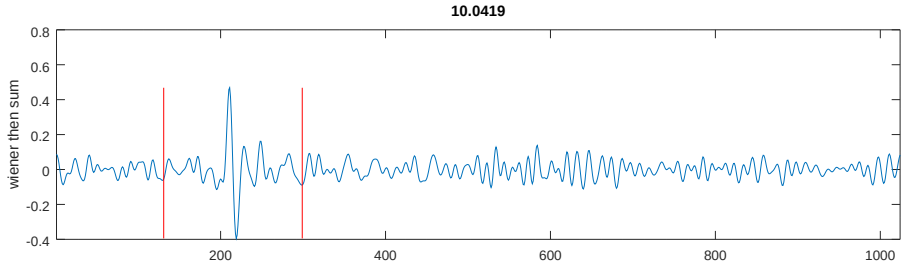
(b) Allpass - Case 2

Figure 29: Comparison between the single-antenna and multi-antenna model applied to z when we apply allpass deconvolution

- Noise level was estimated from signals observed by ANITA when there was no

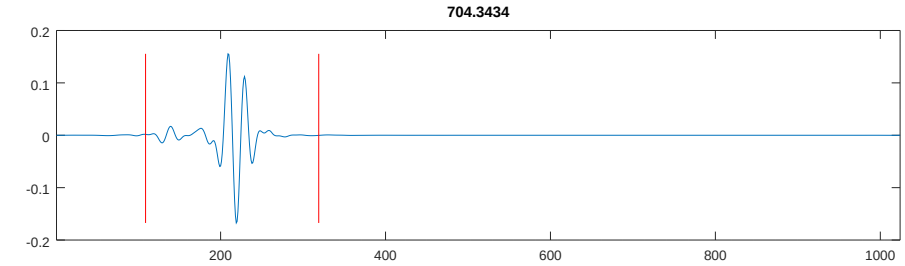


(a) Wiener - Case 1

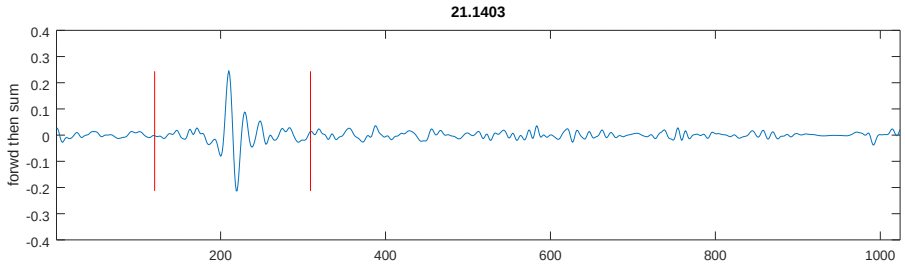


(b) Wiener - Case 2

Figure 30: Comparison between single-antenna and multi-antenna model applied to z when we apply Wiener deconvolution



(a) ForWaRD - Case 1



(b) ForWaRD - Case 2

Figure 31: Comparison between single-antenna and multi-antenna model applied to z when we apply ForWaRD algorithm

distinguishable spike.

- For ANITA data, allpass deconvolution has higher SNR compared to that of Wiener deconvolution. However, ForWaRD produces much better SNR compared to both methods.
- Computing optimal scaling parameters for each signal is computationally expensive so we choose the same set of parameters for each signal, resulting inaccurate thresholding.
- The multi-antenna model is a more accurate representative of the deconvolution problem related to ANITA signals. However, it is more computationally expensive and produces lower SNR compared to that of the single-antenna model. However, the profiles of the output signals are similar to each other.

5.2 Further questions

One of the main obstacles of implementing the deconvolution algorithms is to determine the right noise level present in the observed signal. To deconvolve the ANITA signals, we have used the standard deviation of the noise recorded by the antennas when there were no known signals coming to the antennas. To get a more accurate noise level, we need to record the output of the antennas in isolation. However, we need to take care of other factors (e.g. temperature) that affect the noise level during the ANITA flight while deconvolving the observed signal.

In the context of ANITA signals, we have considered only WAIS pulses, which are known to be localized in time. This helped us define a notion of SNR, even when the original signal is unknown. However, for a general signal (possibly non-localized in time), we do not have a quantity to measure the performance of the deconvolution algorithms. In that case, we have to rely on simulated signals to compare the performances in terms of relative error.

References

[1] Anwei Chai and Zuowei Shen. Deconvolution: A wavelet frame approach. *Numerische Mathematik*, 106(4):529–587, 2007.

[2] Albert Cohen, Marc Hoffmann, and Markus Reiss. Adaptive wavelet galerkin methods for linear inverse problems. *SIAM Journal on Numerical Analysis*, 42(4):1479–1501, 2004.

[3] David L Donoho. De-noising by soft-thresholding. *IEEE transactions on information theory*, 41(3):613–627, 1995.

[4] David L. Donoho and Iain M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.

[5] Michael W. Frazier. *An Introduction to Wavelets Through Linear Algebra*. 1999.

[6] A. D. Hiller and R. T. Chin. Iterative wiener filters for image restoration. In *International Conference on Acoustics, Speech, and Signal Processing*, pages 1901–1904 vol.4, April 1990.

[7] Iain M Johnstone, Gérard Kerkyacharian, Dominique Picard, and Marc Raimondo. Wavelet deconvolution in a periodic setting. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(3):547–573, 2004.

[8] Ramesh Neelamani, Hyeokho Choi, and Richard Baraniuk. Forward: Fourier-wavelet regularized deconvolution for ill-conditioned systems. *IEEE Transactions on signal processing*, 52(2):418–433, 2004.