Multichannel spectral factorization results

Kaiwen Wang and Jon Claerbout

ABSTRACT

Results, finally. hooray

INTRODUCTION

Good morning everyone. The multichannel structure of Figure 4 arises in diverse physical settings.

UNDERSTANDING PEF IN MACHINE LEARNING SETTING

We derive the scalar PEF in machine learning setting and generalize it to the multi-component case.

A scalar case

Assume the observed seismogram y is produced by convolution between a source function s and a time series of spikes x. Our goal is to recover $s(t)$ and $x(t)$ given $y(t)$ as input.

$$y(t) = s(t) * x(t) \tag{1}$$

If we set the length of s to 4, expand equation (1) gives

$$\begin{aligned}
y(t) &= s(1)x(t) + s(2)x(t-1) + s(3)x(t-2) + s(4)x(t-3) \\
y(t-1) &= s(1)x(t-1) + s(2)x(t-2) + s(3)x(t-3) + s(4)x(t-4) \\
y(t-2) &= s(1)x(t-2) + s(2)x(t-3) + s(3)x(t-4) + s(4)x(t-5) \\
y(t-3) &= s(1)x(t-3) + s(2)x(t-4) + s(3)x(t-5) + s(4)x(t-6)
\end{aligned} \tag{2}$$

$y(t)$ could be written in the form of a linear combination of $y(t-1), y(t-2), y(t-3)$

$$
\begin{aligned}
\frac{1}{s(1)}y(t) &= x(t) + \frac{s(2)}{s(1)}x(t-1) + \frac{s(3)}{s(1)}x(t-2) + \frac{s(4)}{s(1)}x(t-3) \\
&= x(t) + \frac{s(2)}{s(1)}(\frac{1}{s(1)}y(t-1) - \frac{s(2)}{s(1)}x(t-2) - \frac{s(3)}{s(1)}x(t-3) - \frac{s(4)}{s(1)}x(t-4)) \\
&\quad + \frac{s(3)}{s(1)}x(t-2) + \frac{s(4)}{s(1)}x(t-3) \\
&= x(t) + \frac{s(2)}{s(1)}\frac{1}{s(1)}y(t-1) + (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})(\frac{1}{s(1)}y(t-2) \\
&\quad - \frac{s(2)}{s(1)}x(t-3) - \frac{s(3)}{s(1)}x(t-4) - \frac{s(4)}{s(1)}x(t-5)) \\
&\quad + (\frac{s(4)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(3)}{s(1)})x(t-3) - \frac{s(2)}{s(1)}\frac{s(4)}{s(1)}x(t-4) \\
&= x(t) + \frac{s(2)}{s(1)}\frac{1}{s(1)}y(t-1) + (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{1}{s(1)}y(t-2) \\
&\quad + (\frac{s(4)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(3)}{s(1)} - (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(2)}{s(1)})x(t-3) \\
&\quad - (\frac{s(2)}{s(1)}\frac{s(4)}{s(1)} + (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(3)}{s(1)})x(t-4) \\
&\quad - (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(4)}{s(1)}x(t-5) \\
&= x(t) + \frac{s(2)}{s(1)}\frac{1}{s(1)}y(t-1) + (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{1}{s(1)}y(t-2) \\
&\quad + (\frac{s(4)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(3)}{s(1)} - (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(2)}{s(1)})(\frac{1}{s(1)}y(t-3) \\
&\quad - \frac{s(2)}{s(1)}x(t-4) - \frac{s(3)}{s(1)}x(t-5) - \frac{s(4)}{s(1)}x(t-6)) \\
&\quad - (\frac{s(2)}{s(1)}\frac{s(4)}{s(1)} + (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(3)}{s(1)})x(t-4) - (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(4)}{s(1)}x(t-5) \\
&= x(t) + \frac{s(2)}{s(1)}\frac{1}{s(1)}y(t-1) + (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{1}{s(1)}y(t-2) \\
&\quad + (\frac{s(4)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(3)}{s(1)} - (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(2)}{s(1)})\frac{1}{s(1)}y(t-3) \\
&\quad - (\frac{s(2)}{s(1)}\frac{s(4)}{s(1)} + (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(3)}{s(1)} \\
&\quad + (\frac{s(4)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(3)}{s(1)} - (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(2)}{s(1)})\frac{s(2)}{s(1)})x(t-4) \\
&\quad - ((\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(4)}{s(1)} + (\frac{s(4)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(3)}{s(1)} - (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(2)}{s(1)})\frac{s(3)}{s(1)})x(t-5) \\
&\quad - (\frac{s(4)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(3)}{s(1)} - (\frac{s(3)}{s(1)} - \frac{s(2)}{s(1)}\frac{s(2)}{s(1)})\frac{s(2)}{s(1)})\frac{s(4)}{s(1)}x(t-6)
\end{aligned}
\tag{3}
$$

Let $y(t) = s(1)x(t) + w(1)y(t-1) + w(2)y(t-2) + w(3)y(t-3) + res$. s' is s scaled by $s(1)$, that is $[1, \frac{s(2)}{s(1)}, \frac{s(3)}{s(1)}, \frac{s(4)}{s(1)}]$. Then we have

$$\begin{aligned} s'(2) &= w(1)s'(1) \\ s'(3) &= w(2)s'(1) + w(1)s'(2) \\ s'(4) &= w(3)s'(1) + w(2)s'(2) + w(1)s'(3) \end{aligned} \tag{4}$$

In general form,

$$s'(n) = \sum_{i=1}^{n-1} w(i)s'(n-i) \tag{5}$$

The residual is

$$\begin{aligned} res = &-(w(1)s'(4) + w(2)s'(3) + w(3)s'(2))s(1)x(t-4) \\ &-(w(2)s'(4) + w(3)s'(3))s(1)x(t-5) \\ &-w(3)s'(4)s(1)x(t-6) \end{aligned} \tag{6}$$

To compare the residual terms with the $s(1)x(t)$ term, we consider a simple example that $s(t) = e^{-kt}$. Then we have $s' = [1, e^{-k}, e^{-2k}, e^{-3k}]$, and $w = [e^{-k}, 0, 0]$. The ratio between the residual terms with the $s(1)x(t)$ term is

$$\frac{res}{s(1)x(t)} = -e^{-4k}\frac{x(t-4)}{x(t)} \tag{7}$$

This shows that for exponential decay source function, the residual terms are negligible compared with $s(1)x(t)$. Neglecting the residual terms, $y(t)$ could be written as $y(t) = s(1)x(t) + w(1)y(t-1) + w(2)y(t-2) + w(3)y(t-3)$, in general,

$$y(t) = s(1)x(t) + w(1)y(t-1) + w(2)y(t-2) + \cdots + w(n)y(t-n), \tag{8}$$

where the length of the source function is $n+1$.

In a scalar case, we use linear regression to make prediction in a time series $y(t)$. Our assumption is that each data point could be predicted by a linear combination of its previous data points. Then prediction error is $y(t) - (w(1)y(t-1) + w(2)y(t-2) + \cdots + w(n)y(t-n)) = s(1)x(t)$. By doing linear regression, we achieve blind deconvolution that satisfies $y(t) = x(t) * s(t) = [s(1)x(t)] * s'(t)$.

As shown in Figure 1, $y^{(i)} = x(t+n)$ is predicted by $w^T x^{(i)}$, where $x^{(i)}$ is a preceding data segment $[x(t), x(t+1), x(t+2), \cdots, x(t+n-1)]$ of length n. The loss function L is set to be l_2 norm of the prediction error.

$$L(w) = \|y^{(i)} - \hat{y}^{(i)}\| = (y^{(i)} - w^T x^{(i)})^2 \tag{9}$$

We calculate the gradient of loss function at each time step,

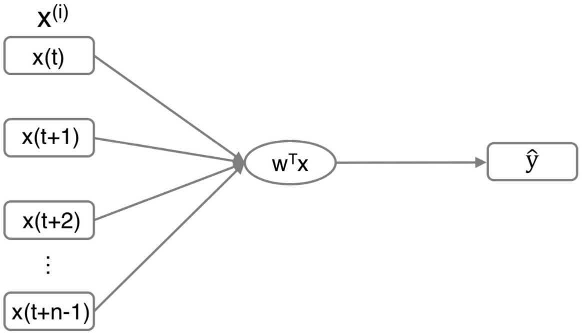

Figure 1: Diagram of scalar PEF. $x^{(i)}$ is a data segment beginning at time t. $\hat{y}$ is the prediction.

$$\frac{dL(w)}{dw} = -2(y^{(i)} - w^T x^{(i)})x^{(i)} \tag{10}$$

Then we apply gradient descent at each time step and update the parameter vector w using the gradient. α is a chosen learning rate.

$$w = w - \alpha \frac{dL(w)}{dw} \tag{11}$$

By running the update with time, we will get a prediction error vector. The prediction error $y - \hat{y}$ is a time series of length m. The physical meaning is unpredictable part of $x(t)$.

If we assume stationary source function, we can apply batch gradient descent, which pass the entire time series in each iteration. w^T here will be time invariant. We define cost function J as the mean of loss function at each training data$(x^{(i)}, y^{(i)})$,

$$J(w) = \frac{1}{m} \sum_{i=1}^{m} \|y^{(i)} - \hat{y}^{(i)}\| = \sum_{i=1}^{m} (y^{(i)} - w^T x^{(i)})^2 \tag{12}$$

The gradient of cost function is

$$\frac{dJ(w)}{dw} = -2 \sum_{i=1}^{m} (y^{(i)} - w^T x^{(i)})x^{(i)} \tag{13}$$

Applying gradient descent update will give prediction error $y - \hat{y}$.

multicomponent case

Figure 2 shows how two different sources contribute to recordings at a two-component instrument. The Green's function and source time function are both unknowns. x_1 and x_2 are our observations that are two-component seismograms. Our goal is to perform blind deconvolution to recover the Green's function and source time function.

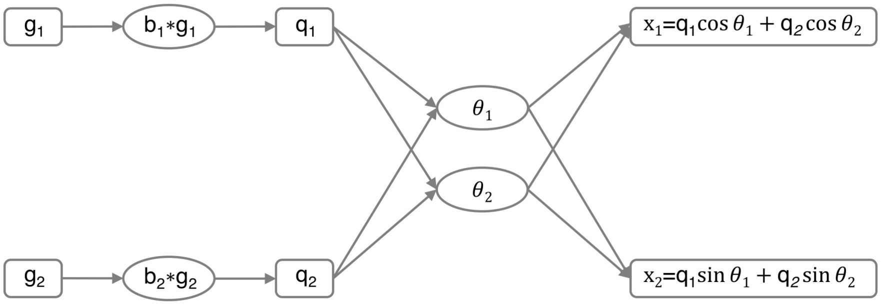

Figure 2: Diagram of how an earthquake generates multicomponent recordings. g_1 and g_2 denote Green's function of the two sources. Convolution of the Green's function and source time function b_1 and b_2 gives q_1 and q_2, as the recordings along the particle motion direction, respectively. Then the two waves q_1 and q_2 are projected to the horizontal and vertical directions.

The network we present is shown in Figure 3. We propose that an inverse rotation should be the first step to remove the dependence between traces. This is an inverse of the projection step on the right of Figure 2. In this step, parameter θ_1 and θ_1 could be either time variant or invariant, and is learned through gradient updates. The forward relationship in Figure 2 is

$$\begin{aligned} x_1 &= q_1 \cos\theta_1 + q_2 \cos\theta_2, \\ x_2 &= q_1 \sin\theta_1 + q_2 \sin\theta_2 \end{aligned} \tag{14}$$

The inverse is then

$$\begin{aligned} q_1 &= \frac{x_1 \sin\theta_2 - x_2 \cos\theta_2}{\cos\theta_1 \sin\theta_2 - \sin\theta_1 \cos\theta_2}, \\ q_2 &= \frac{x_1 \sin\theta_1 - x_2 \cos\theta_1}{\cos\theta_2 \sin\theta_1 - \sin\theta_2 \cos\theta_1} \end{aligned} \tag{15}$$

After the inverse rotation, q_1 and q_2 are inputted into a linear regression step. Now the problem reduces to a scalar problem as we discussed in the previous section. The loss function is defined as

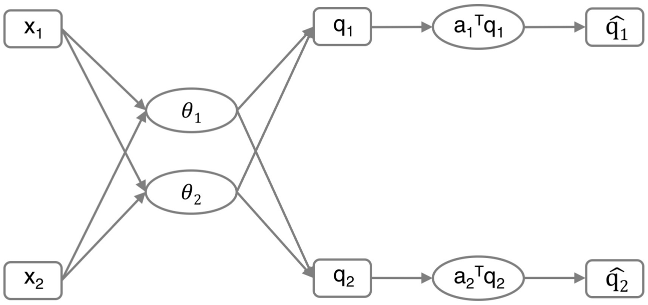

Figure 3: The network for deconvolution. x_1 and x_2 are multicomponent recordings as the input of the network. They first go through an inverse rotation step to rotate back to the directions of their own particle motions. After the inverse rotation, they could be treated as two separate scalar cases, where the two set of parameters a_1 and a_2 are learned independently.

$$L(a_1, a_2, \theta_1, \theta_2) = \|q_1^{(i)} - \hat{q}_1^{(i)}\| + \|q_2^{(i)} - \hat{q}_2^{(i)}\| = (q_1(t+n) - a_1^T q_1^{(i)})^2 + (q_2(t+n) - a_2^T q_2^{(i)})^2 \tag{16}$$

Then we apply gradient descent at each time step and update the parameters $a_1, a_2, \theta_1, \theta_2$ using the gradient. α is a chosen learning rate.

$$\begin{aligned} a_1 &= a_1 - \alpha \frac{dL}{da_1}, \\ a_2 &= a_2 - \alpha \frac{dL}{da_2}, \\ \theta_1 &= \theta_1 - \alpha \frac{dL}{d\theta_1}, \\ \theta_2 &= \theta_2 - \alpha \frac{dL}{d\theta_2} \end{aligned} \tag{17}$$

The gradients are calculated as follows,

$$
\begin{aligned}
&\frac{dL}{da_1} = -2(q_1(t+n) - a_1^T q_1^{(i)})q_1^{(i)},\\
&\frac{dL}{da_2} = -2(q_2(t+n) - a_2^T q_2^{(i)})q_2^{(i)},\\
&\frac{dL}{d\theta_1} = \frac{dL}{dq_1^{(i)}}\frac{dq_1^{(i)}}{d\theta_1} + \frac{dL}{dq_2^{(i)}}\frac{dq_2^{(i)}}{d\theta_1} = -2(q_1(t+n) - a_1^T q_1^{(i)})a_1^T\frac{dq_1^{(i)}}{d\theta_1} - 2(q_2(t+n) - a_2^T q_2^{(i)})a_2^T\frac{dq_2^{(i)}}{d\theta_1},\\
&\frac{dL}{d\theta_2} = \frac{dL}{dq_1^{(i)}}\frac{dq_1^{(i)}}{d\theta_2} + \frac{dL}{dq_2^{(i)}}\frac{dq_2^{(i)}}{d\theta_2} = -2(q_1(t+n) - a_1^T q_1^{(i)})a_1^T\frac{dq_1^{(i)}}{d\theta_2} - 2(q_2(t+n) - a_2^T q_2^{(i)})a_2^T\frac{dq_2^{(i)}}{d\theta_2}
\end{aligned}
\tag{18}
$$

If we want to separate P and S waves, then we have approximately $\theta_1 + \theta_2 = \frac{\pi}{2}$. Set $\theta_1 = \theta, \theta_2 = \frac{\pi}{2} - \theta$, the forward and backward relationship reduces to

$$
\begin{aligned}
x_1 &= q_1 \cos\theta + q_2 \sin\theta,\\
x_2 &= q_1 \sin\theta + q_2 \cos\theta
\end{aligned}
\tag{19}
$$

The inverse is then

$$
\begin{aligned}
q_1 &= \frac{x_1 \cos\theta - x_2 \sin\theta}{\cos 2\theta},\\
q_2 &= \frac{x_2 \cos\theta - x_1 \sin\theta}{\cos 2\theta}
\end{aligned}
\tag{20}
$$

TEST CASES

test case with mixing but no filtering

test case with filtering

test case of random spikes

Stripped of details, ℓ_2-norm scalar-signal code is

```
a(1) = 1.0           #               Syntax:   "a+=b" means "a=a+b"
do for all time t    # e = nonstationary prediction error
        do tau= 1, na
                       e(t)    +=  a(tau) * y(t-tau+1)      # forward
        do tau= 2, na
                       da(tau) +=  e(t)   * y(t-tau+1)      # adjoint
        do tau= 2, na
                       a = a - epsilon * da
```

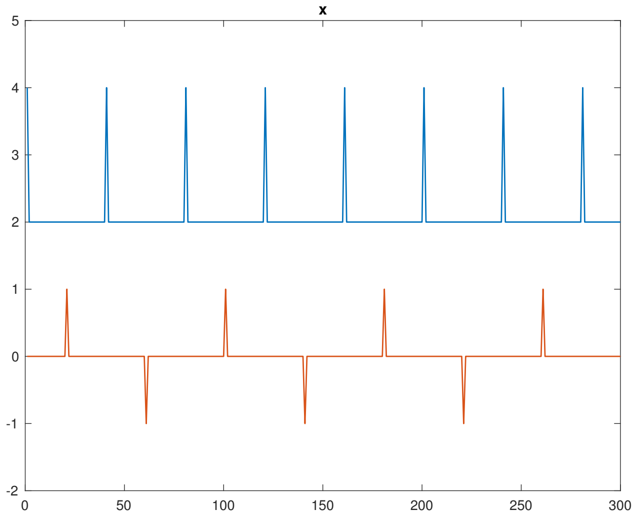

Figure 4: Unmixed traces $\mathbf{x}$ as the ground truth after the deconvolution process. $\mathbf{x}_1$ is designed to have spikes of same size and interval. In $\mathbf{x}_2$ the spikes changed in polarity.

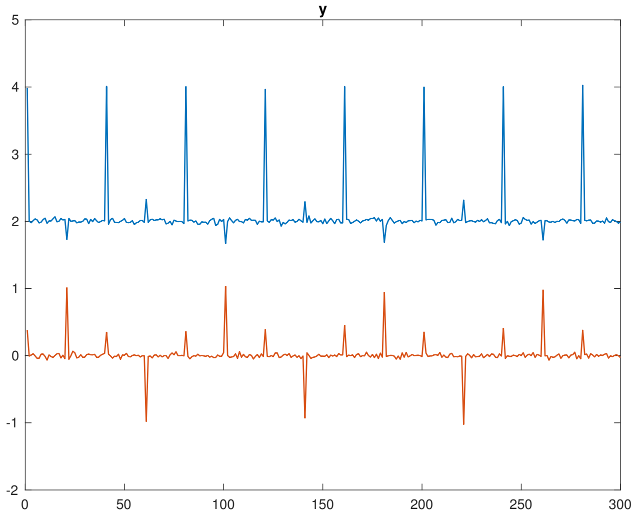

Figure 5: We applied a filter $\mathbf{B} = \left(\begin{smallmatrix} 1 & -0.3 \\ 0.2 & 1 \end{smallmatrix}\right)$ on $\mathbf{x}$ and add Gaussian noise to obtain $\mathbf{y}$ as our observations on horizontal and vertical components.

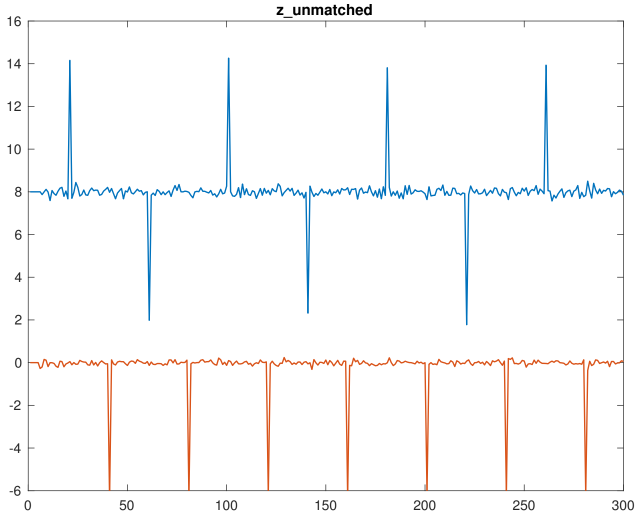

Figure 6: **z** is the output of deconvolution. Compared with ground truth the traces may change in order and polarity.

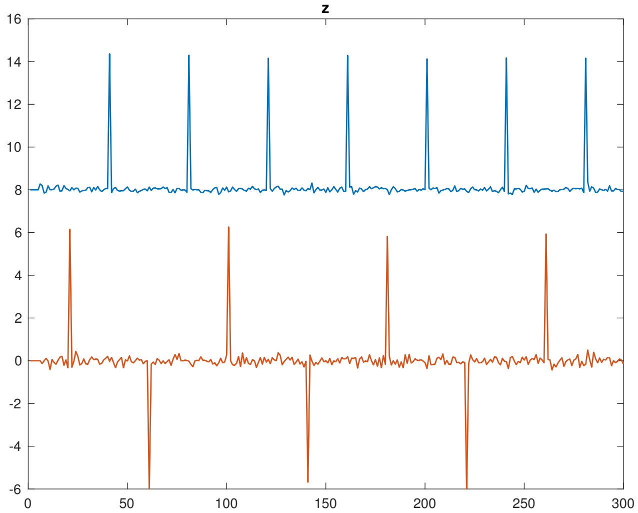

Figure 7: Adding constrain that z_1 is more similar to y_1 than y_2 and z_2 is more similar to y_2 than y_1, we define the polarity and order of the outputs.

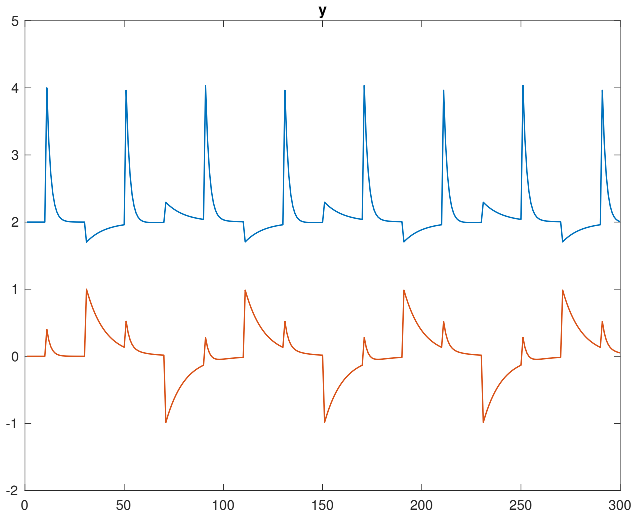

Figure 8: The traces **x** is designed the same as the first test case. A filter $\mathbf{B} = \begin{pmatrix} 1/(1{-}0.6Z) & -0.3/(1{-}0/9Z) \\ 0.2/(1{-}0/6Z) & 1/(1{-}0/9Z) \end{pmatrix}$ is applied to generate **y**. Synthetic data input is vertical and horizontal components. Model is a mix of sharp, unipolar P waves and S waves of lower frequency with alternating polarity. Stronger P waves on the vertical, and stronger S waves on the horizontal.

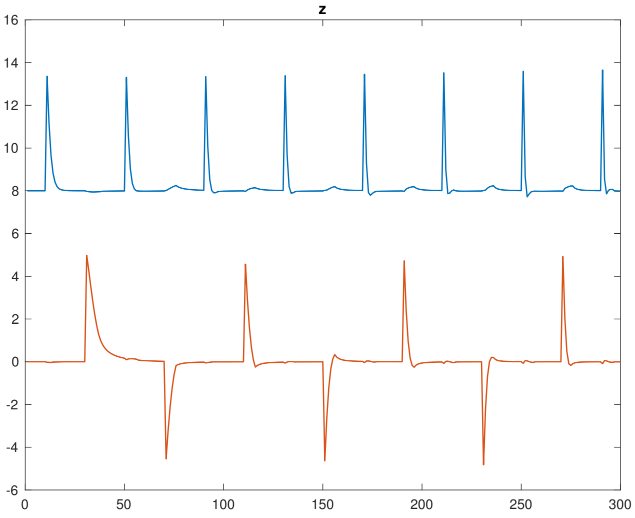

Figure 9: Outputs of deconvolution have P wave on vertical component and S on horizontal component. Spiking improves with time.

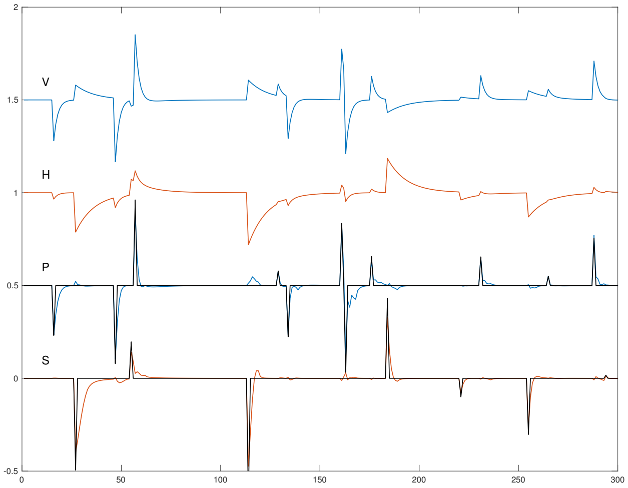

Figure 10: Top two traces are vertical and horizontal observation. Blue and red curves on the bottom are deconvolution results overlaid by ground truth spikes in black

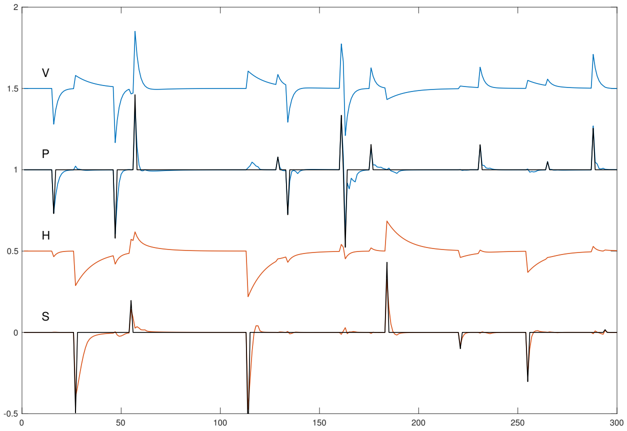

Figure 11: The traces are the same as Figure 10 except that they are reordered. Top two traces are vertical observation and deconvolution result which contains P wave spikes. Bottom two traces are horizontal observation and deconvolution result which contains S wave spikes.

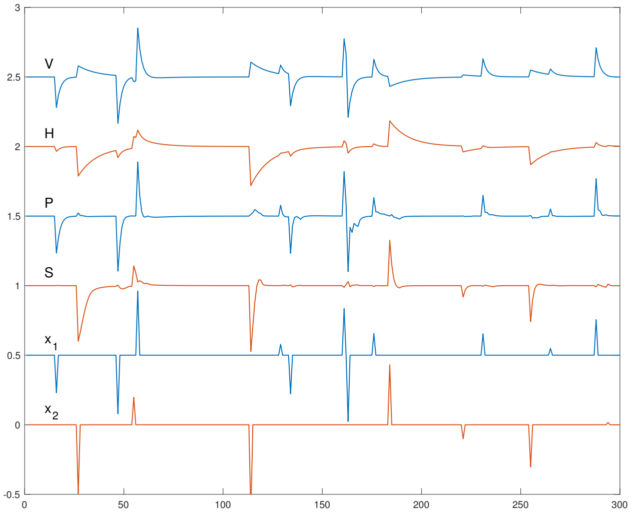

Figure 12: Top two traces are vertical and horizontal observation. Following two are deconvolution results and bottom two traces are original spikes.

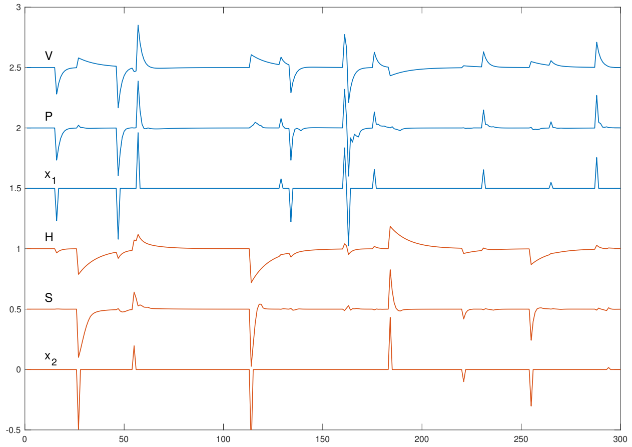

Figure 13: The traces are the same as Figure 12 except that they are reordered. Top three traces are vertical observation, deconvolution result and true P wave spikes. Bottom three traces are horizontal observation, deconvolution result and true S wave spikes.

TEST CASES

To compare inputs with outputs, display as 50 channels per sheet of wiggle trace in 10 groups of 5, the 5th being a dead trace to clarify display.

Easiest case, mixing, but no filtering

```
polarity = 1.
do i=1,1000,40  {   # jump in steps of size 40
      x1(i) = 2.
      x2(i+20) = polarity
      polarity = -polarity
      }
```

$$\mathbf{B} \quad = \quad \begin{bmatrix} 1 & -.3 \\ .2 & 1 \end{bmatrix} \tag{21}$$

The unscrambling would all be done by Cholesky and Unitary.

A case with an obvious answer

I'm not sure the method of this paper should unscramble it. I'm not really sure what these methods should be capable of, but this one should be really impressive if it works.

$$\mathbf{B} \quad = \quad \begin{bmatrix} 1./(1-.6Z) & -.3/(1-.9Z) \\ .2/(1-.6Z) & 1./(1-.9Z) \end{bmatrix} \tag{22}$$

For more fun, we might prefer wavelets that oscillate.

Horizontal phones, two far away signals, near each other, one stronger, the other 5× *weaker coming in at slightly different angles*

$$\mathbf{x} \quad = \quad \begin{bmatrix} S \\ W \end{bmatrix} \quad = \quad \begin{bmatrix} \text{random numbers} \\ \text{random numbers} \end{bmatrix} \qquad \text{for i} = 1, 10000 \tag{23}$$

$$\mathbf{B} \quad = \quad \begin{bmatrix} 5/(1-.6Z) & 1/(1-.9Z) \\ 4/(1-.6Z) & 1/(1-.9Z) \end{bmatrix} \tag{24}$$

Consistent wavelet polarities

Greg Beroza reminds us of repeating earthquakes. Especially small quakes may repeat with the same polarity. We should think about whether and how such phenomena can be best recognized.

Somewhere we should refer to a famous book **?**.

We'll be referring to the paper (**?**)