Multi-scale Full Waveform Inversion of Acoustic Approximation

Fuqiang Chen

Earth Science and Engineering
KAUST

Outline

Implementation of FWI

Explicitly re-written using Matrix Notation

Numerical tests in 1D medium

FWI and Multi-scale FWI

Conclusion and Further study

MATLAB Code

Implementation of FWI

Forward Problem: $$p(\mathbf{x}_r, t) = \int d\mathbf{x}_s G(\mathbf{x}_r, t; \mathbf{x}_s) * s(\mathbf{x}_s, t)$$

If one source and one receiver configuration is considered, then the sum over the numbers of shots and receivers are removed.

Using Matrix notations, the misfit function can be re-written as

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \vdots \\ \delta_{(nt-1)} \\ \delta_{nt} \end{bmatrix}^T \begin{bmatrix} g_1 & & & & & \\ g_2 & g_1 & & & & \\ g_3 & g_2 & g_1 & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ g_{(nt-1)} & g_{(nt-2)} & g_{(nt-3)} & \cdots & g_1 & \\ g_{nt} & g_{(nt-1)} & g_{(nt-2)} & \cdots & g_2 & g_1 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ \vdots \\ s_{(nt-1)} \\ s_{nt} \end{bmatrix}$$

The middle Matrix acts as the process of convolution

Implementation of FWI

Change the order of inner product calculation and combine the matrix with residual waveform.

$$\begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ \vdots \\ s_{(nt-1)} \\ s_{nt} \end{bmatrix}^T \begin{bmatrix} g_1 & g_2 & g_3 & \cdots & g_{(nt-1)} & g_{(nt)} \\ & g_1 & g_2 & \cdots & g_{(nt-2)} & g_{(nt-1)} \\ & & g_1 & \cdots & g_{(nt-3)} & g_{(nt-2)} \\ & & & \ddots & \vdots & \vdots \\ & & & & g_1 & g_2 \\ & & & & & g_1 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \vdots \\ \delta_{(nt-1)} \\ \delta_{nt} \end{bmatrix}$$

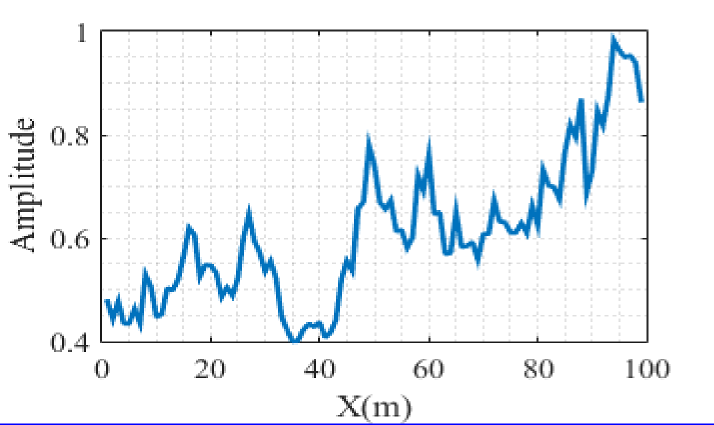
Now, the convolution operator acts as cross-correlation operator without negative lag.
We can re-write the cross-correlation operator into convolution by reverse residual field.

$$\begin{bmatrix} g_1 & & & & & \\ g_2 & g_1 & & & & \\ g_3 & g_2 & g_1 & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ g_{(nt-1)} & g_{(nt-2)} & g_{(nt-3)} & \cdots & g_1 & \\ g_{nt} & g_{(nt-1)} & g_{(nt-2)} & \cdots & g_2 & g_1 \end{bmatrix} \begin{bmatrix} \delta_{nt} \\ \delta_{(nt-1)} \\ \delta_{(nt-2)} \\ \vdots \\ \delta_2 \\ \delta_1 \end{bmatrix}$$

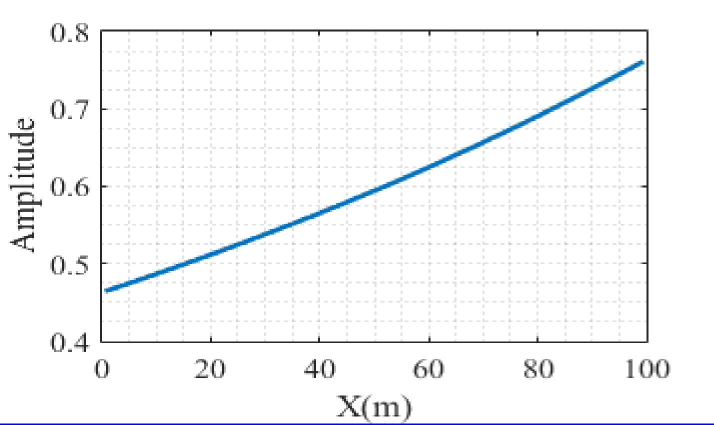
To calculate misfit function, we should make the time sequence of forward and backward wave field consistent, it is easy to do by propagating forward wave field from maximum time to zero time.

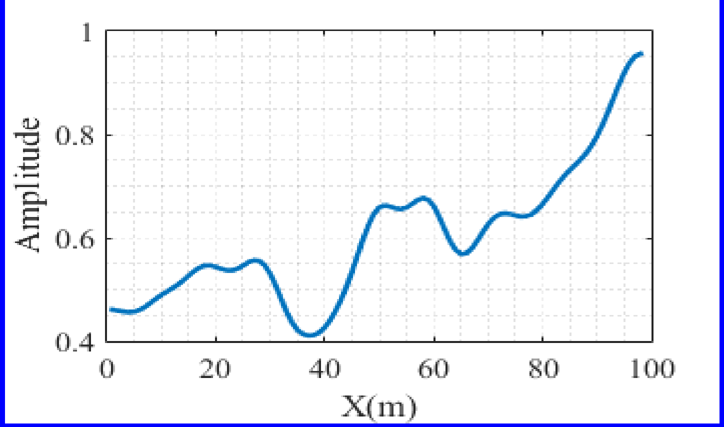
Numerical tests in 1D medium

True Model

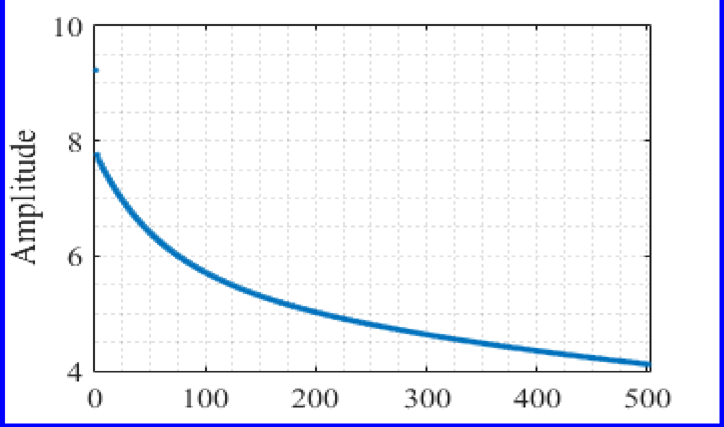


Initial Model





Inverted Model

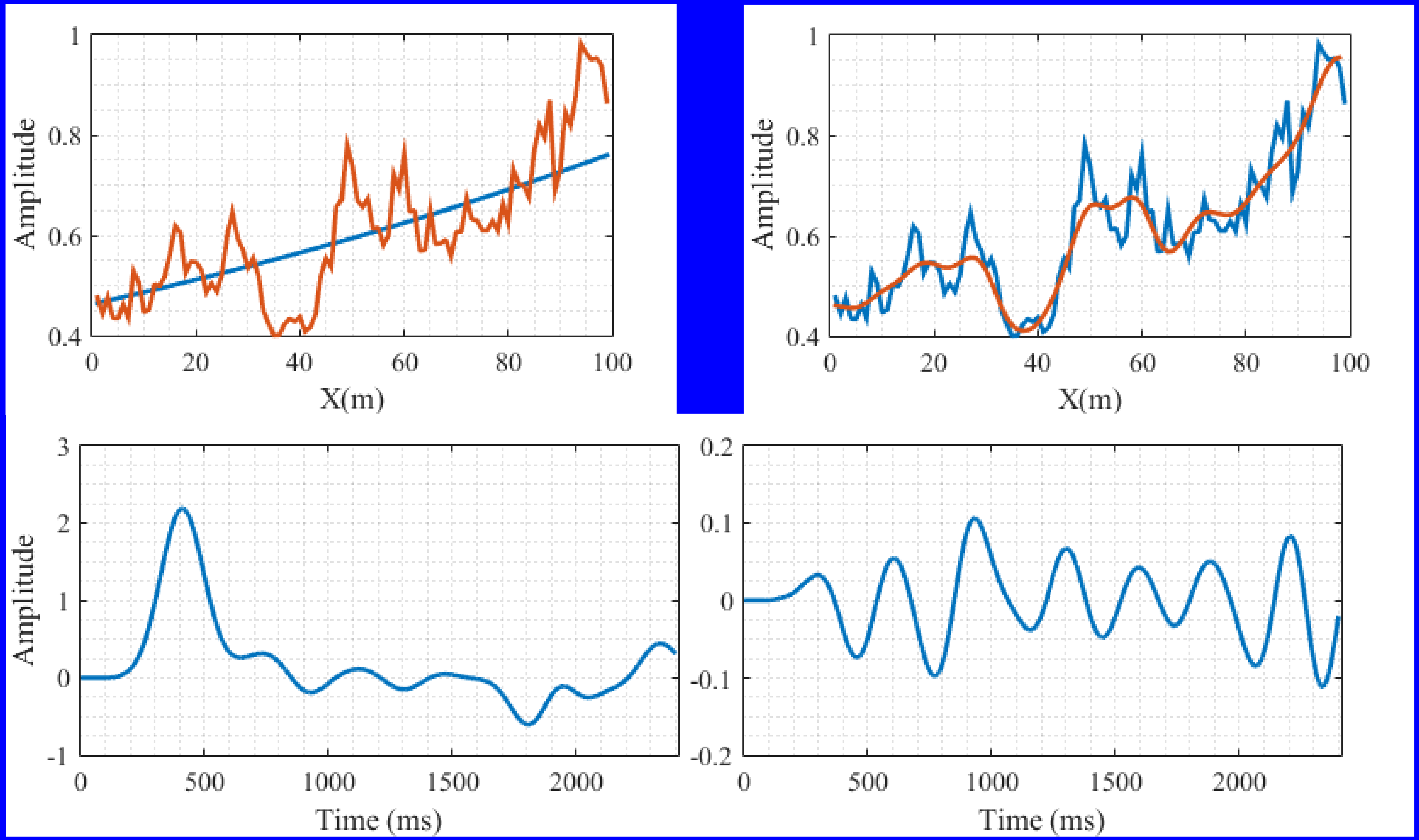


Iteration Curve

Numerical tests in 1D medium

Overlaid True Model vs Initial Model (a) and True Model vs Inverted Model (b)

(a) (b)



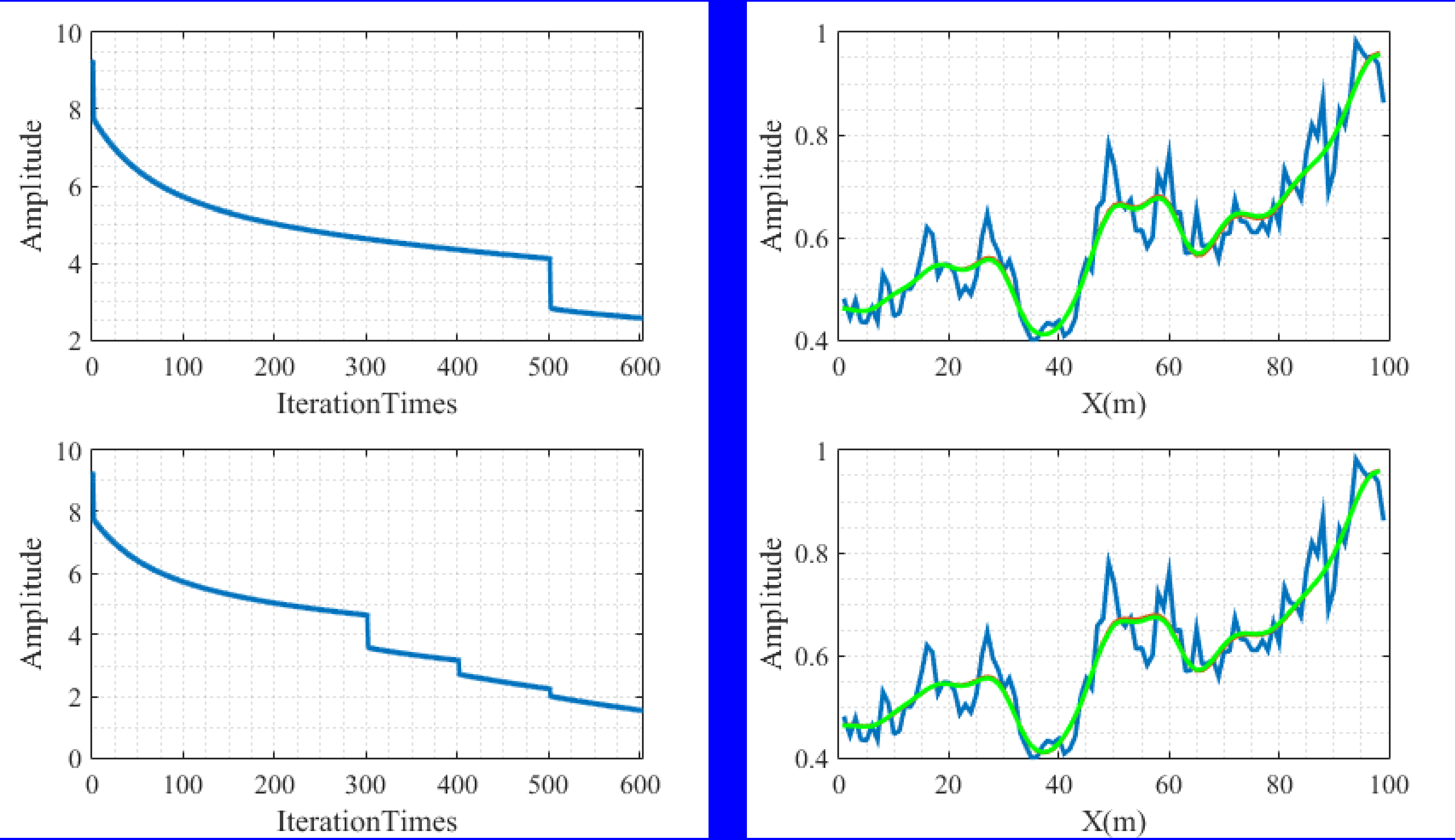
(c) (d)

Observed Data (c) and Data Residual (d)

Numerical tests in 1D medium

Smoothing Residual using 1 stage: Iteration Curve (a) and Overlaid Velocity Model (b)

(a) (b)

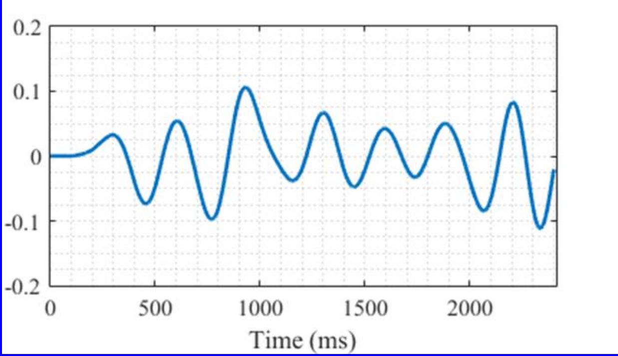


(c) (d)

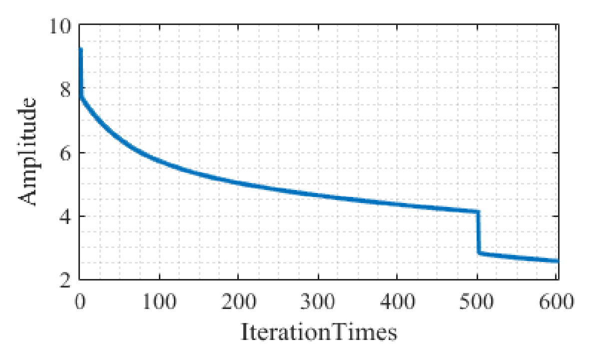
Smoothing Residual using 3 stages: Iteration Curve (c) and Overlaid Velocity Model (d)

Numerical tests in 1D medium

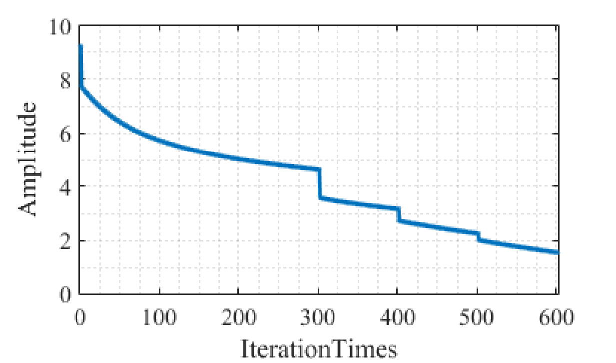
(a)



(b)



(c)



Residual data after 501 iterations (a), Iteration curve relates to smooth residual after 501st iteration and iteration curve relates to smooth residual data at 301st, 401st and 501st.

Because observed data is dominated by low frequency and narrow frequency range, It seems that multi scale scheme does not work very well.

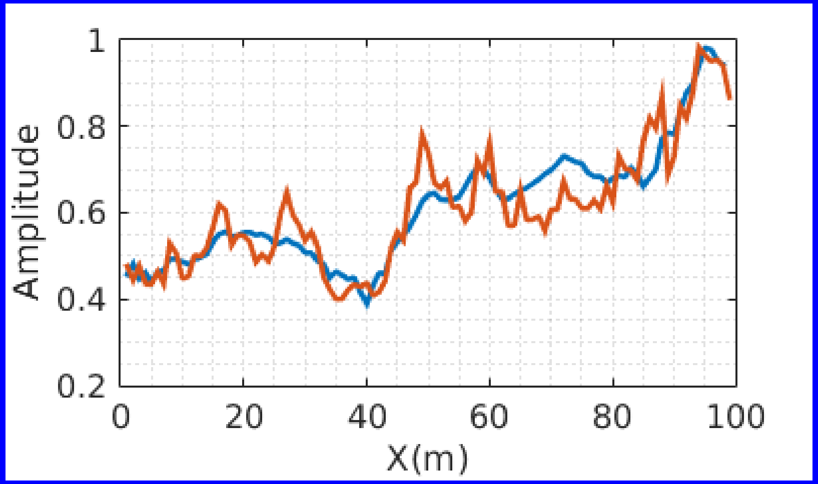
Figure (a) shows that the residual data contains high frequency components much more than observed data. So we can smooth the residual data to continue the inversion process.

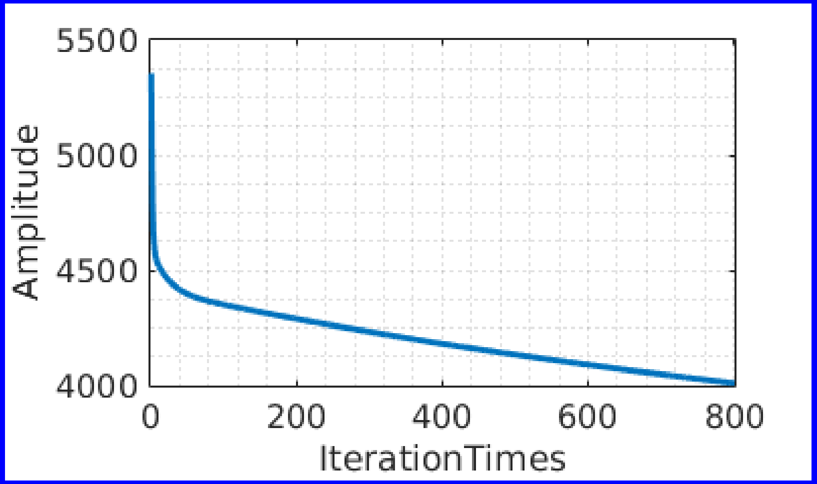
Both figure (c) and (d) show that the misfit functions decrease further, while figure (c) is minimized much than figure (b).

Numerical tests in 1D medium

Inverted model (a) from data with highly dominant frequency(c) and iteration curve (b).

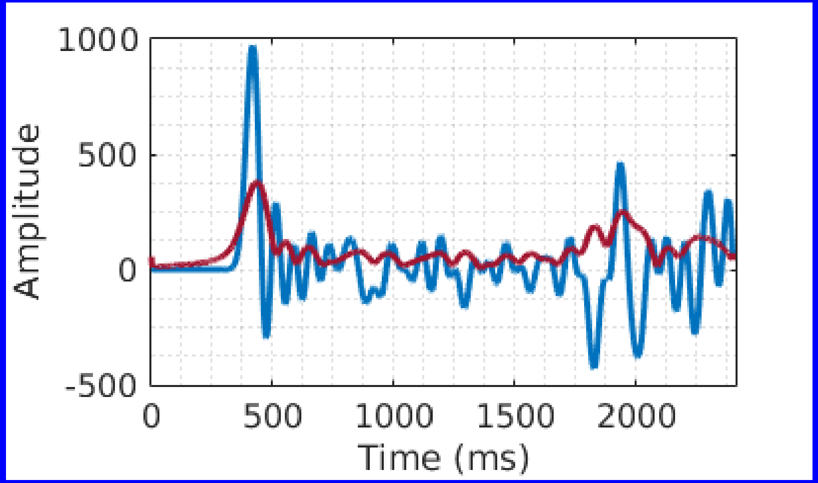
(a)





(b)

(c)

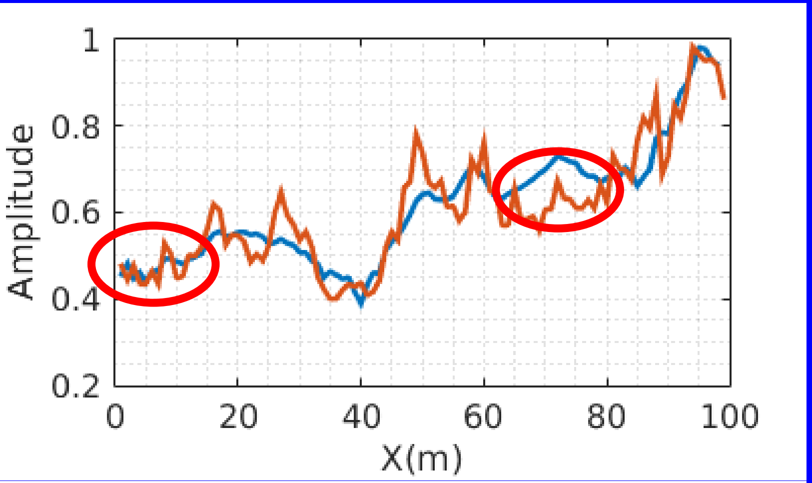


To adapt the data for multi scale scheme,
the wavelet of higher frequency is used.
But it seems that the results do not improve
Obviously. Maybe it is because the frequency
range is narrow.

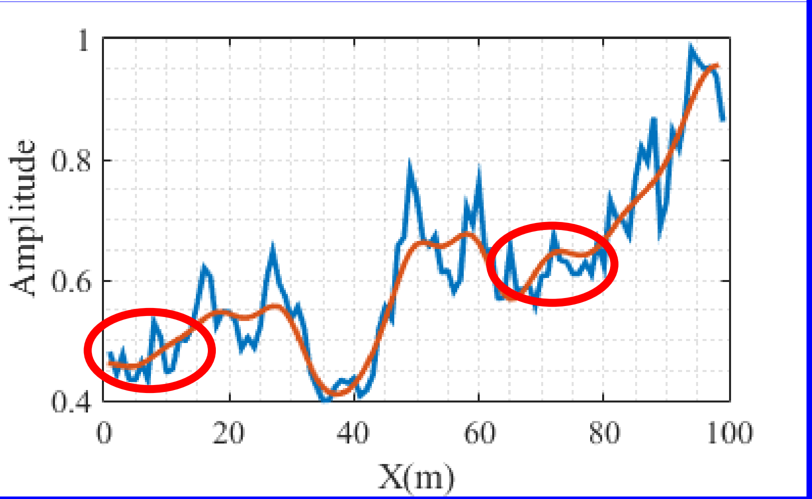
Numerical tests in 1D medium

Inverted results from Data dominated by low frequency (a) and dominated by high frequency (b)

(a)



(b)



Conclusion: It seems that inversion procedure converges faster for data dominated by low frequency than data dominated by high frequency. In this tests, the result of figure (a) needs 801 times iteration while the result of figure (b) Needs 501 times iteration.

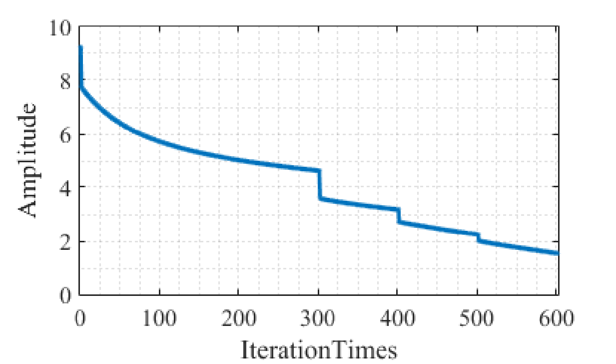
b> Inverted resulted from data dominated by high frequency shows higher apparent resolution than figure (b) does. But which resolution causes ambiguous and, maybe need to be smoothed out for the next use. In other words, results from high frequency is less reliable than the results from low frequency.

MATLAB Code

```
if it > 301 && it <= 401
    datares(rec,:)=sgolayfilt(datares(rec,:), 3, 501);
end

if it > 401 && it <= 501
    datares(rec,:)=sgolayfilt(datares(rec,:), 3, 601);
end

if it > 501
    datares(rec,:)=sgolayfilt(datares(rec,:), 3, 701);
end
```



Multi-scale of Data Residual, the procedure of iteration is divided by 5 different smoothing weights.

MATLAB Code

```
 for it = tmin+1 : tmax-2
        for ix = xmin+1 : xmax-1
            p(i,it+1)=2*p(ix,it)-p(ix,it-1)+ …
                    (c(ix)*dt)^2*(S(ix,it)+(p(ix+1,it)-2*p(ix,it)+p(ix-1,it))/dx^2);
        end
        p(xmin,it+1)=0.;
        p(xmax,it+1)=p(xmax-1,it+1);
    end
    for it = tmin+2:tmax-1
        datares(rec,it)=(p(rec,it)-Pobs(rec));
    end
end
```

Forward modelling and residual of waveform calculates

MATLAB Code

```
for it = tmax-1:-1:tmin+2
       for ix = xmin+1:xmax-1
           delta(ix,it-1)=2*delta(ix, it)-delta(ix, it+1) + …
                   (c*dt)^2*(datares(ix,it)+…
             (delta(i+1,a)-2*delta(ix, it)+delta(i-1,a))/dx^2);
       end
       delta(xmin,it-1)=0.;
       delta(xmax,it-1)=delta(xmax-1,it-1);
end

for it = tmin+1:tmax-2
      for ix = xmin+1:xmax-1
           dp2(ix,it) = (p(ix,it+1)-2*p(ix,it) + p(ix,it-1))/dt^2;
      end
end
```

The second derivative of wave field and reverse propagation of virtual source.