Title: “Computing the vertical density and seismic velocity profiles from multi-angle reflection data: error analysis”

Date: Student name: W.Y. Meijer
ABSTRACT

Numerical models were used to recursively compute the density profile and the seismic velocity profile of three different artificial models of the underground from primary reflection images obtained from multi-angle incident plane wave reflection data. The aim is to investigate the effect of errors in the obtained primary reflection amplitudes on the recursive construction of the vertical density-velocity profiles. The reflection coefficients needed for the computations were obtained by solving the Marchenko equation for different angles of incidence. The recursive computation shows errors occur in every layer, but the error does not necessarily grow with each step. This implies the error does not propagate into the recursive scheme. Adding a random error to the reflection coefficients yielded results in a greater error in each individual layer with respect to the values obtained without an added error. Using a too large angle of incidence can result in too few primary events in the autofocused data, distorting the computed values.
CONTENTS

Abstract........................................................................................................................................................................

1. Introduction .............................................................................................................................................................. 1

2. Reflection data and “single-sided” autofocusing ................................................................................................. 2
   1.1 Constructing reflection data .......................................................................................................................... 2
   1.2 Auto-focusing .................................................................................................................................................. 4
   1.3 Amplitude Picking ......................................................................................................................................... 5
       Implementation in matlab ............................................................................................................................... 5
   1.4 Determining p and c ..................................................................................................................................... 6
       Implementation in matlab ............................................................................................................................... 6
   1.5 Numerical Models .......................................................................................................................................... 7
       Model 1 ......................................................................................................................................................... 7
       Model 2 ......................................................................................................................................................... 8
       Model 3 ......................................................................................................................................................... 9

3. Results .................................................................................................................................................................. 10
   MODEL 1: Two distinct angles on incidence ....................................................................................................... 10
   MODEL 1: 75 different angles of incidence ....................................................................................................... 11
   MODEL 2: 75 different angles of incidence ....................................................................................................... 12
   MODEL 3: 75 angles of incidence ....................................................................................................................... 14
   MODEL 1: 75 angles and the effect of noise ....................................................................................................... 17

4. Conclusions ............................................................................................................................................................ 19
   4.1 Acknowledgement ......................................................................................................................................... 19

5. Literature ............................................................................................................................................................... 19

6. Appendices ............................................................................................................................................................. 20
   Appendix A: Multipulsesolution.m .................................................................................................................... 20
   Appendix B: RoseVin.m ....................................................................................................................................... 22
   Appendix C: Primypick.m ..................................................................................................................................... 25
   Appendix D: Sincinterpol.m ................................................................................................................................. 26
1. INTRODUCTION.

In reflection seismic acoustic data is gathered to construct an image of the subsurface. Primary reflections are the main interest in this process. Multiple reflections and refracted waves are seen as noise. Commonly stacking, migration and other data processing techniques are used to create an image that can be used for interpretation. Most processing removes the free-surface multiples, while internal multiples are usually ignored during the migration process. Over the last few years, several schemes are designed that remove internal multiples. These schemes depend on a subsurface model and the quality of internal multiple removal depends on the quality of this model. Independently, in acoustics Rose suggested an iterative scheme to focus events at arbitrary one-way travel-time (Rose, 2002). Because the scheme does not need a subsurface model, Rose called the method “single-sided” autofocus. De Waard showed that the iteration scheme works for an normal incident plane wave (Waard, 2011) for an artificially created reflection response, resulting in correctly computed one-way travel-times. He found that a specific implementation of this scheme yields the correct local reflection coefficient at one-way travel-time. This presents a direct linear method to create an internal-multiple free image.

To characterize the subsurface seismic inversion methods are developed, which find the subsurface density and velocity distribution. These methods rely on seismic data amplitude and the medium parameters are found by iterative forward modeling starting from an initial model. These methods pose the inverse problem as an ill-posed non-linear problem. For a horizontally layered earth we can use the image containing only primary reflection coefficients at one-way travel-times to recursively find the density and velocity by starting at the receiver level and the knowledge of the density and velocity in the upper half-space. Using the relation between the acoustic impedance and the reflection coefficients the seismic velocity and density can be calculated. This will have to be done recursively, which could lead to error propagation. The question this report seeks to answer is:

Does error propagation occur by recursively computing density and seismic velocity for a given model using primary reflection events obtained from the autofocus scheme?

In Chapter 2 the theory and numerical methods are discussed for creating a reflection data, retrieving primary reflection coefficients by autofocus, picking local reflection amplitudes, and computing the densities and seismic velocities. This chapter also contains the description of the models used to test the procedure. Chapter 3 will show the results for the different models. The effect of adding noise to the reflection response is also presented here. Chapter 4 contains the conclusion and discussion.
2. Reflection Data and “Single-Sided” Autofocusing

1.1 Constructing Reflection Data

Part of an acoustic wave propagating through a fluid will be reflected back at a boundary between two layers with different physical properties, namely mass density and the propagation velocity. The other part of the wave will be transmitted through the boundary. Part of this transmitted wave will be reflected at the next boundary and another part will pass through and so on, regardless from which side the wave approaches the boundary. The strength of the reflected signal depends on the reflection coefficient (equation 1.1) of the boundary, which depends on the acoustic impedance of the layers on each side.

\[ r_n = \frac{Z_n - Z_{n+1}}{Z_n + Z_{n+1}}, \tag{1.1} \]

in which \( r_n \) is the reflection coefficient and \( Z_n \) an \( Z_{n+1} \) are the acoustic impedances of the two layers.

Consider two finite spaces between two infinite vertical half-spaces. A downward propagating wave, generated at an arbitrary source depth level in the top half-space, will encounter three boundaries. At each boundary part of the wave field will be reflected back up again and the other part will be transmitted downwards. A receiver on the same arbitrary depth level as the source will receive the total wave field that is transmitted back up again. Upward traveling waves that have reflected once are called primaries. The rest of the recorded wave field consists of multiply reflected waves and are called multiples. These are usually ignored or seen as noise. All primary and multiple events together are called the reflection data.

De Waard (Waard, 2011) showed that the reflection response for a downward traveling wave incident on a layered model of the underground can be built. For an \( N \)-layered model a bottom-up recursive formula was used to create the reflection response:

\[ R_n = \frac{r_n + R_{n+1}e^{-2i\omega t_{n+1}}}{1 + r_nR_{n+1}e^{-2i\omega t_{n+1}}}, \tag{1.2} \]

which recursion is started by taking \( R_0=0 \), because this is the bottom half space and there is no reflected wave propagating upwards to the bottom boundary. The model used in this report contains four layers of which the bottom and the top ones are half spaces. Because \( N=4 \), the recursive formula thus goes from \( n=3 \) to \( n=0 \), with \( R_{3}=0 \). To construct the reflection response the reflection coefficients of the different boundaries need to be known. For this equation 1.1 is used but the acoustic impedance still needs to be calculated. The acoustic impedance depends on the angle, \( \alpha \) with which the initial downward propagating wave is incident on the horizontal interface,

\[ Z_n = \frac{\rho_n}{\sqrt{1/c_n^2 - p^2}}, \tag{1.3} \]

\[ p = \frac{\sin(\alpha)}{c_1}, \tag{1.4} \]

in which \( Z_n \) is the acoustic impedance of the \( n^{th} \) layer, \( c_n \) is the seismic velocity and \( \rho_n \) is the density in that layer. When a wave approaches a boundary under an angle the horizontal slowness, \( \rho \), has to be taken into account (equation 1.4). Using the script in appendix A a reflection responses for different angles of incidence can be
generated for the models shown further on in this chapter. The reflection response (Figure 1) contains the primary reflections and multiples.

The script was written by De Waard and could only be used for normally incident waves. To accommodate other angles of incidence, the script was altered as computing the impedance for not normally incident waves is slightly different. Horizontal slowness has to be taken into account. De Waard used the following script to calculate $Z_n$:

$$
\begin{align*}
\text{kap} &= 1 / (\rho \cdot v^2) \\
Z &= \sqrt{\rho / \text{kap}} \\
r &= (Z(1:nl-1) - Z(2:nl)) / (Z(1:nl-1) + Z(2:nl))
\end{align*}
$$

where $Z$ is the acoustic impedance, $\rho$ the density, $v$ the seismic velocity and $r$ the reflection coefficient.

This is correct for a normal incident wave because when $\alpha=0$ the horizontal slowness in equation 1.4 becomes zero and does not have to be taken into account. The wave only propagates downwards and not to the side. To make the script useable for angles other than 0°, equation 1.4 was implemented into the script. This resulted in the following lines of script replacing those mentioned before:

$$
\begin{align*}
p &= \sin(\alpha) / v(1) \\
slo &= \sqrt{v(-2) - p^2} \\
\text{kap} &= \rho \cdot slo^2 \\
Z &= \sqrt{\text{kap} / \rho} \\
r &= (Z(1:nl-1) - Z(2:nl)) / (Z(1:nl-1) + Z(2:nl))
\end{align*}
$$

The two parameters added are $p$ and $slo$, where $p$ stands for the horizontal slowness, while $slo$ represents the vertical slowness. Only when the vertical slowness is a non-zero real number the autofocusing scheme will work, because when the vertical slowness becomes zero critical reflection and refraction will take place resulting in non-unique travel paths in the subsurface, while when it becomes imaginary evanescent waves occur in the data that are not described by the autofocusing scheme.

FIGURE 1) UNFOCUSED DATA 20 DEG ANGLE OF INCIDENCE ON MODEL 1
1.2 Auto-focusing

To determine the densities and the seismic velocities an image containing the primary reflections with the local reflection amplitudes at the one-way travel time are needed. As can be seen in Figure 1, the response signal constructed in the previous contains not only the primary reflections but also the multiples. In the ideal situation for the model used in this report would take the form of,

\[ I(t) = r_0 \delta(t - t_0) + r_1 \delta(t - t_0 - t_1) + r_2 \delta(t - t_0 - t_1 - t_2). \]  \hspace{1cm} (2.1)

Rose suggests in his article (Rose, 2002) a prescription for autofocusing the response signal in order to find the primary reflections. The advantage is that no equations need to be solved. Density and seismic velocity are not required to use this method. It relies solely on measured input and output to focus the data to retrieve only the primary reflections. Rose suggests the following steps:

“Step 1
Interrogate the sample with the nth incident pulse, \( \varphi_{in}(t - x/c_0; t_f) \), that commences with a sharp right-going unipolar pulse that crosses the origin at \( t = -t_f \) and is followed by any smooth trailing wave, a ‘tail’. See figure 1(a) for an example incident pulse. In the limit, the sharp unipolar pulse becomes the delta function \( \delta(t + t_f - x/c_0) \).

Step 2
Record the reflected left-going pulse on the left half-line to obtain \( \varphi_{in}(t + x/c_0; t_f) \).

Step 3
Evaluate the result at \( x = 0 \) and truncate for \( t > t_f : \varphi_{in}(t; t_f) \Theta(t_f - t) \).

Step 4
Time reverse the result, \( t \rightarrow -t \), to obtain \( \varphi_{in}(-t; t_f) \Theta(-t + t_f) \).

Step 5
Subtract this result from \( \delta(t + t_f) \) to obtain \( \delta(t + t_f) - \varphi_{in}(-t; t_f) \Theta(t + t_f) \).

Step 6
Let \( t \rightarrow t - x/c_0 \). The result is the new right-going incident pulse

\[ \varphi_{in}(t - x/c_0; t_f) = \delta(t - x/c_0 + t_f) - \varphi_{in}(-t + x/c_0; t_f) \Theta(t - x/c_0 + t_f); \]

Step 7
Iterate until convergence.

Here, \( \Theta(x) \) equals one if \( x \geq 0 \) and zero otherwise” (Rose, 2002).

De Waard (Waard, 2011) used the single sided autofocusing to successfully find the primary reflections at one-way travel time for three models, which he built using the procedure described in the previous chapter using a normal incident pulse. The primary reflections can also be found when dealing with an incident pulse. An adjusted form of De Waard script is used to autofocus the data (Figure 12) obtained from the models in this report.

The adjustments in the scripts are small and focus on the fact that multiple angles of incidence are needed to calculate the density and seismic velocity. De Waard used two different scripts one that contains all the parameters needed to create a reflection response and another that creates the reflection response and does the autofocus procedure. In the previous section the adjustments put in place to create a response with a wave that travels under an angle of incidence is already discussed. The adjustment to the combined script is, that it is turned into a MATLAB-function file. The function file needs four parameters in order to perform the computation: angle of incidence, density model, seismic velocity model and a depth model. One of the output parameters of the script is the sequence of reflection coefficients, which were used to create the reflection response. These are used for comparison with the reflection coefficients obtained from the autofocusing
process. The rest of the output data are a time vector, autofocused response signal and the ideal response signal, which are needed to reconstruct the input data.

1.3 Amplitude Picking
The auto-focusing described in the previous chapter renders a new signal with only the primary reflections in the time domain. In order to determine the density and seismic velocity the amplitude of each reflection has to be determined. The one-way travel time is only of interest once velocity has been determined because it allows for a time depth conversion.

The algorithm used to obtain the amplitudes of the primary reflection makes use of the fact that only the before mentioned reflection differs from zero when looking at the ideal situation (equation 2.1). The algorithm starts at the beginning of the signal by checking the following condition and works its way towards the end.

$$|R(t_n)| < a. (3.1)$$

Where $a$ is a threshold value below which the signal $R(t)$ is not considered to be part of a reflection. This bandwidth allows for a smooth running of the numerical approach as $R(t)$ will not always equal zero besides the reflections, because of bandwidth limitations. As long as the condition in equation 3.1 is met the algorithm moves on to the next point of the trace without recording anything. Once equation 3.1 is not valid the algorithm records its position and moves on a new WHILE-loop where the following statement has to be upheld:

$$|R(t_n)| \geq a \text{ or } |R(t_{n+1})| \geq a. (3.2)$$

The end point of the reflection is recorded once condition 3.2 is not met any more. Checking $R(t_{n+1})$ ensures the algorithm stays in the reflection loop until the end of a particular reflection. It is possible that a reflection will contain both a positive and a negative deflection from zero. If this is the case and a point in the reflection $|R(t_n)|$ is smaller than $a$ then the loop would be terminated while the reflection is not recorded in its entirety.

Once the start and the end time of the reflection are found this part of the signal is extracted and run through the sincinterpol algorithm developed by Jürg Hunziker (appendix D) This algorithm transforms the reflection to the frequency domain in order to get a higher resolution.

Implementation in MATLAB
After the true reflection response is autofocused a reflection response $R$ (in matlab called trace) is retrieved. A WHILE loop is used to check the values in the reflection response. Equation 3.1 in matlab becomes:

```matlab
while abs(trace(n))<=0.01;
    n=n+1;
end
```

Once the condition is not true anymore the value of $n$ is stored and a new while loop is entered. This time the condition is as stated in equation 3.2.

```matlab
l(i,1)=n;
while abs(trace(n))>0.01 || abs(trace(n+1))>0.01;
    n=n+1;
end
```

At a certain point this statement is not valid and the value of $n$ is once again stored. In the next step the part of the reflection response containing the reflector can be extracted and run through sincinterpol.m. After this is
done the maximum divergence from 0 is extracted. A FOR-loop ensures that the script starts at the first condition again. The entire script is looped as many times as there are reflectors.

1.4 Determining ρ and C

The density and the seismic velocity in the top half-space are assumed known. Given this fact, the densities and seismic velocities can be computed recursively when the reflection coefficients are available. The amplitude found in the previous chapter has the same value as the reflection coefficient of that particular boundary.

The relation between the acoustic impedance of two and the reflection coefficient of the boundary in between these is as follows.

\[ R_n = \frac{Z_n - Z_{n+1}}{Z_n + Z_{n+1}} \] (4.1)

which can be rewritten as,

\[ Z_{n+1} = Z_n \frac{1 - R_n}{1 + R_n} \] (4.2)

The relation between the acoustic impedance, seismic velocity and the density of a medium is given by the following equation as in chapter 1.

\[ Z_n = \frac{\rho_n}{\sqrt{1/c_n^2 - p^2}} \] (4.3)

Combining equations 4.1 and 4.2 gives a relation between the impedance of the first layer, the horizontal slowness (p), the density of the next layer and the seismic velocity in the next layer. Resulting in an equation where only the density and seismic velocity are unknown.

\[ \frac{1}{c_{n+1}^2} - \rho_{n+1}^2 \left( Z_n \frac{1 - R_n}{1 + R_n} \right)^{-2} = p^2 \] (4.4)

where the horizontal slowness (p) is given by equation 1.4.

The horizontal slowness can be computed using the angle of incidence and the seismic velocity in the first layer. The result is that equation 4.4 has two unknowns and the horizontal slowness depends on the angle of incidence. Using two angles of incidence would result in two equations with two unknowns. This can be solved exactly. If more than two angles of incidence are used to find reflection coefficients the entire procedure leads to an over determined system, which can be solved using the least-squares method. Once the density and the seismic velocity have been calculated for the second layer the same equations can be used to compute the acoustic parameters of the third layer, and so on.

Implementation in MATLAB

Obtaining the values for the density and seismic velocity for each layer in the model can be done recursively. The known values of the top layer can be used to calculate the values of the second layer and so on. As the number of reflectors, N-1, are known from the model the script has to loop N-1 times to retrieve all the values. Using the fact that the density and the seismic velocity in the top layer are known, equation 4.3 is implemented in matlab as:
\[
Z(n,:) = \left( \frac{\rho(n)^2}{\sqrt{\left(\frac{1}{(c(n,:)^2)} - p\right)}} \right)
\]

Where \( n \) goes from 1 to the number of reflectors present in the model. To calculate impedance for all the angles at once ‘;' is used. The horizontal slowness is calculated beforehand, as it does not change from layer to layer. Equation 4.5 becomes:

\[
p = \left(\frac{\sin(\alpha)}{c(1)}\right)^2;
\]

Now equation 4.2 can be used to determine the impedance for the next layer.

\[
Z(n+1,:) = Z(n,:) \cdot \left(\frac{1-r(n,:)}{1+r(n,:)}\right);
\]

As \( Z(n+1) \) is now described in terms \( Z(n) \), which is known, the system of equations can be built into a matrix, \( A \), of \( k \) times 2, where \( k \) is number of angles of incidence. The first column is filled entirely by 1’s and the second column is filled with second half of the left part of equation 4.4 except for \( \rho^2 \) because that is what needs to be calculated. As there is now an over determined system we apply the least square solution. The values obtained are for \( 1/c^2 \) and \( \rho^2 \) in a vector with length of two. Rewriting these results in the values for the density and the seismic velocity. The entire process is repeated for the next layer by means of a for-loop.

1.5 NUMERICAL MODELS

To see if density and seismic velocity could be accurately recovered from the signal response after “single-sided” autofocusing three models were used. The models contained respectively four, seven and nine layers. Each model contains two halfspaces at the top and the bottom. All of them contained a dummy interface in the top halfspace at which the reflection response was recorded. Figure 2, Figure 4 and Figure 6 show a schematic picture of the three different models with the heights, densities and seismic velocities of all the layers for each model. In all the models the velocity fluctuates from layer to layer.

**MODEL 1**

<table>
<thead>
<tr>
<th>( c_0 )</th>
<th>( \rho_0 )</th>
<th>( h_0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1500 m/s</td>
<td>2000 kg/m³</td>
<td>( \infty )</td>
</tr>
</tbody>
</table>

--- Dummy Interface ---

<table>
<thead>
<tr>
<th>( c_0 )</th>
<th>( \rho_0 )</th>
<th>( h_0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1500 m/s</td>
<td>2000 kg/m³</td>
<td>175 m</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( c_1 )</th>
<th>( \rho_1 )</th>
<th>( h_1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2500 m/s</td>
<td>2100 kg/m³</td>
<td>150 m</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( c_2 )</th>
<th>( \rho_2 )</th>
<th>( h_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1600 m/s</td>
<td>1900 kg/m³</td>
<td>140 m</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( c_3 )</th>
<th>( \rho_3 )</th>
<th>( h_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000 m/s</td>
<td>1900 kg/m³</td>
<td>( \infty )</td>
</tr>
</tbody>
</table>

**FIGURE 2) MODEL 1**

A wave propagating through the model under an angle creates a reflection response as can be seen in Figure 3. In this case the angle of incidence is 5°. De Waard’s script convolves the data with a Ricker wavelet. The first peak corresponds to the first boundary, between the top halfspace and the layer underneath. The second
corresponds to the third boundary. The third peak is a multiple while the fourth peak corresponds to the third boundary.

**FIGURE 3) MODEL 1 REFLECTION RESPONSE 5°G ANGLE OF INCIDENCE**

**MODEL 2**

<table>
<thead>
<tr>
<th>$c_0$</th>
<th>$\rho_0$</th>
<th>$h_0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1500 m/s</td>
<td>2000 kg/m$^3$</td>
<td>$\infty$</td>
</tr>
<tr>
<td>1500 m/s</td>
<td>2000 kg/m$^3$</td>
<td>200 m</td>
</tr>
<tr>
<td>2500 m/s</td>
<td>2100 kg/m$^3$</td>
<td>150 m</td>
</tr>
<tr>
<td>1600 m/s</td>
<td>1900 kg/m$^3$</td>
<td>200 m</td>
</tr>
<tr>
<td>2000 m/s</td>
<td>1900 kg/m$^3$</td>
<td>175 m</td>
</tr>
<tr>
<td>1800 m/s</td>
<td>2200 kg/m$^3$</td>
<td>150 m</td>
</tr>
<tr>
<td>2500 m/s</td>
<td>2100 kg/m$^3$</td>
<td>140 m</td>
</tr>
<tr>
<td>1700 m/s</td>
<td>2300 kg/m$^3$</td>
<td>$\infty$</td>
</tr>
</tbody>
</table>

**FIGURE 4) MODEL 2**
There are six reflectors in this model. The reflection response only shows five large peaks, which correspond to primary reflections (Figure 5). A sixth primary reflection is obscured by the multiples in the reflection response of a wave with an angle of incidence of 5°.

![Graph showing convolved with Ricker wavelet](chart.png)

**FIGURE 5) MODEL 2 REFLECTION RESPONSE 5DEG ANGLE OF INCIDENCE**

**MODEL 3**

<table>
<thead>
<tr>
<th>c_0</th>
<th>1500 m/s</th>
<th>( \rho_0 = 2000 \text{ kg/m}^3 )</th>
<th>( h_0 = \infty )</th>
</tr>
</thead>
<tbody>
<tr>
<td>c_1</td>
<td>2500 m/s</td>
<td>( \rho_1 = 2100 \text{ kg/m}^3 )</td>
<td>( h_1 = 100 \text{ m} )</td>
</tr>
<tr>
<td>c_2</td>
<td>1600 m/s</td>
<td>( \rho_2 = 1900 \text{ kg/m}^3 )</td>
<td>( h_2 = 200 \text{ m} )</td>
</tr>
<tr>
<td>c_3</td>
<td>2000 m/s</td>
<td>( \rho_3 = 1900 \text{ kg/m}^3 )</td>
<td>( h_3 = 150 \text{ m} )</td>
</tr>
<tr>
<td>c_4</td>
<td>1800 m/s</td>
<td>( \rho_4 = 2200 \text{ kg/m}^3 )</td>
<td>( h_4 = 200 \text{ m} )</td>
</tr>
<tr>
<td>c_5</td>
<td>2500 m/s</td>
<td>( \rho_5 = 2100 \text{ kg/m}^3 )</td>
<td>( h_5 = 175 \text{ m} )</td>
</tr>
<tr>
<td>c_6</td>
<td>1700 m/s</td>
<td>( \rho_6 = 2300 \text{ kg/m}^3 )</td>
<td>( h_6 = 150 \text{ m} )</td>
</tr>
<tr>
<td>c_7</td>
<td>1950 m/s</td>
<td>( \rho_7 = 2000 \text{ kg/m}^3 )</td>
<td>( h_7 = 140 \text{ m} )</td>
</tr>
<tr>
<td>c_8</td>
<td>2100 m/s</td>
<td>( \rho_8 = 1900 \text{ kg/m}^3 )</td>
<td>( h_8 = \infty )</td>
</tr>
</tbody>
</table>

**FIGURE 6) MODEL 3**
Figure 7 shows the reflection response of model 3. It shows a lot of similarities with the reflection response of model although a different number of layers were used. It contains some smaller peaks.

![convolved with Ricker wavelet](image)

**FIGURE 7) MODEL 3 REFLECTION RESPONSE 5DEG ANGLE OF INCIDENCE**

### 3. RESULTS

In this chapter the results using two distinct angles of incidence are tested to see if there is error propagation when the method described in chapter 2 is used. This is tested on model 1. Using more data points would in theory increase the accuracy of the result. To test this 75 angles of incidence, evenly spread from 0° to 10° less than the critical angle (equation 6.1) of incidence, are used. To see how well the method holds when more layers are used the same is done for model 2 and 3. At the end model 1 is used again to see what happens if the signal contains noise. To test this, first a maximum random error of 5% is added or subtracted from the obtained reflection coefficients. This was done a second time with a maximum error of 10%.

\[
\sin(\alpha) = \frac{c_n}{c_{n+1}}. \quad (6.1)
\]

The critical angle is given by \( \alpha \). In this case the wave is traveling downwards \( c_n \) denotes the velocity in the layer above the boundary and \( c_{n+1} \) the velocity in the layer beneath the boundary. If the wave is propagating upwards it is the other way around.

**MODEL 1: TWO DISTINCT ANGLES ON INCIDENCE**

Using the velocity profile from model 1 and putting the values for the first and second layer into equation 6.1 results in a critical angle of 36.87°. So two angles lower than approximately 25° are used. Angles of 5° and 20° are chosen for this first attempt to obtain the velocity and density profile from the autofocusing data. Table 1 shows that the reflection coefficients obtained after autofocusing are not the same as the reflection coefficients used to create the initial reflection response. In the case of the first and the third boundaries the reflection responses obtained after autofocusing are lower than the original while for the second reflective boundary they are higher.
Table 2 shows the original input values and the obtained values for the velocity and density of each of the four layers. The velocity is an overestimate of the true values and the density is underestimated. However using two distinct angles of incidence does not result in an increase in the relative error with each recursive step. As the deviation for both the velocity and the density in layer 4 show a smaller relative error than those of layer 3.

MODEL 1: 75 DIFFERENT ANGLES OF INCIDENCE
The same model as applied in the previous section is used. The critical angle of incidence is therefore still 36.87°. To keep a margin from the critical angle of incidence the angles were chosen from 0° to 25° equidistance apart from each other. The reflection coefficients (Figure 8) show an upward moving trend the closer the angle of incidence is to the critical angle. The reflection coefficients for each reflector and angle are either
Table clearly shows that as in the previous section the relative error does not increase with each recursive step. The speed has a maximum deviation of 0.15 while the density is 0.09 off (Table 3). For both the velocity and the density the greatest error occurs in layer 3.

**TABLE 3) RESULTS MODEL 1 75 ANGLES OF INCIDENCE**

<table>
<thead>
<tr>
<th>layer</th>
<th>input c (m/s)</th>
<th>output c (m/s)</th>
<th>rel. err. (-)</th>
<th>input ρ (kg/m3)</th>
<th>output ρ (kg/m3)</th>
<th>rel. err (-)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1500</td>
<td>1500</td>
<td>0.00</td>
<td>2000</td>
<td>2000</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>2500</td>
<td>2525</td>
<td>0.01</td>
<td>2100</td>
<td>2054</td>
<td>-0.02</td>
</tr>
<tr>
<td>3</td>
<td>1600</td>
<td>1832</td>
<td>0.15</td>
<td>1900</td>
<td>1726</td>
<td>-0.09</td>
</tr>
<tr>
<td>4</td>
<td>2000</td>
<td>2161</td>
<td>0.08</td>
<td>1900</td>
<td>1811</td>
<td>-0.05</td>
</tr>
</tbody>
</table>

**MODEL 2: 75 DIFFERENT ANGLES OF INCIDENCE**

As the velocity in the top halfspace and the layer underneath are the same as in model 1 the critical angle of incidence is also the same. The angles in this case also range from 0° to 25°. This is also convenient for comparing the results with those of model 1 as the same amount and the same angles are used to obtain the results. Unfortunately the script ran into an error after an angle of 23.2° angle because the amplitude-picking algorithm could not find all the reflection coefficients.

To see if results could be obtained using multiple angles of incidence the maximum angle was decreased to 20°. Figure 9 shows that all the reflection coefficients are present in the range from 0° to 20° angle of incidence. This resulted in the values shown in Table 4. Just as for model 1 all the calculated velocity are lower than the original input values while all the densities are overestimated in comparison to the original values. The error does not increase when the velocity and the density for each underlying layer is calculated. Looking at the numbers no clear pattern is discernible. Plotting the relative error for the velocity and the density on the y-axis and the layer number on the x-axis it does seem like there could be an increasing trend (Figure 10).
TABLE 4) RESULTS MODEL 2 FOR 75 ANGLES OF INCIDENCE

<table>
<thead>
<tr>
<th>layer</th>
<th>input c (m/s)</th>
<th>output c (m/s)</th>
<th>rel. err. (-)</th>
<th>input ρ (kg/m³)</th>
<th>output ρ (kg/m³)</th>
<th>rel. err. (-)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1500</td>
<td>1500</td>
<td>0.00</td>
<td>2000</td>
<td>2000</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>2500</td>
<td>2488</td>
<td>0.00</td>
<td>2100</td>
<td>2134</td>
<td>0.02</td>
</tr>
<tr>
<td>3</td>
<td>1600</td>
<td>1326</td>
<td>-0.17</td>
<td>1900</td>
<td>2192</td>
<td>0.15</td>
</tr>
<tr>
<td>4</td>
<td>2000</td>
<td>1812</td>
<td>-0.09</td>
<td>1900</td>
<td>2023</td>
<td>0.06</td>
</tr>
<tr>
<td>5</td>
<td>1800</td>
<td>1566</td>
<td>-0.13</td>
<td>2200</td>
<td>2444</td>
<td>0.11</td>
</tr>
<tr>
<td>6</td>
<td>2500</td>
<td>2353</td>
<td>-0.06</td>
<td>2100</td>
<td>2131</td>
<td>0.01</td>
</tr>
<tr>
<td>7</td>
<td>1700</td>
<td>1232</td>
<td>-0.28</td>
<td>2300</td>
<td>2954</td>
<td>0.28</td>
</tr>
</tbody>
</table>

FIGURE 9) REFLECTION COEFFICIENTS REFLECTORS OF MODEL 2

FIGURE 10) RELATIVE ERRORS DENSITY AND SEISMIC VELOCITY FOR MODEL 2
MODEL 3: 75 ANGLES OF INCIDENCE

Again the maximum angle of incidence taken is 25°. The results are shown in Table 5. Especially from layer 5 onwards the errors are large and range between 0.69 and 1.27 for the velocity and between 0.45 and 0.61 for the density. The different reflection coefficients show jump after the 5th reflector. All of them follow the line they should (Figure 11), but then they suddenly jump and start following the true reflection coefficient line of another reflector.

TABLE 5) RESULTS MODEL 3 75 ANGLES OF INCIDENCE

<table>
<thead>
<tr>
<th>layer</th>
<th>input c (m/s)</th>
<th>output c (m/s)</th>
<th>rel. err. (-)</th>
<th>input ρ (kg/m³)</th>
<th>output ρ (kg/m³)</th>
<th>rel. err (-)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1500</td>
<td>1500</td>
<td>0.00</td>
<td>2000</td>
<td>2000</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>2500</td>
<td>2525</td>
<td>0.01</td>
<td>2100</td>
<td>2055</td>
<td>-0.02</td>
</tr>
<tr>
<td>3</td>
<td>1600</td>
<td>1831</td>
<td>0.14</td>
<td>1900</td>
<td>1727</td>
<td>-0.09</td>
</tr>
<tr>
<td>4</td>
<td>2000</td>
<td>2160</td>
<td>0.08</td>
<td>1900</td>
<td>1812</td>
<td>-0.05</td>
</tr>
<tr>
<td>5</td>
<td>1800</td>
<td>3043</td>
<td>0.69</td>
<td>2200</td>
<td>1213</td>
<td>-0.45</td>
</tr>
<tr>
<td>6</td>
<td>2500</td>
<td>3974</td>
<td>0.59</td>
<td>2100</td>
<td>825</td>
<td>-0.61</td>
</tr>
<tr>
<td>7</td>
<td>1700</td>
<td>3866</td>
<td>1.27</td>
<td>2300</td>
<td>821</td>
<td>-0.64</td>
</tr>
<tr>
<td>8</td>
<td>1950</td>
<td>3846</td>
<td>0.97</td>
<td>2000</td>
<td>1093</td>
<td>-0.45</td>
</tr>
<tr>
<td>9</td>
<td>2100</td>
<td>3918</td>
<td>0.87</td>
<td>1900</td>
<td>779</td>
<td>-0.59</td>
</tr>
</tbody>
</table>

FIGURE 11) REFLECTION COEFFICIENTS MODEL 3 (0 DEG TO 25 DEG ANGLE OF INCIDENCE)
In this case there seems to be a mismatch between what the reflection coefficients should be and the ones found. On closer inspection of the autofocused data at an angle of incidence of 5°, where the input reflection coefficients match those found, and 25°, where there is a mismatch, show some interesting results. As can be seen from Figure 12 and Figure 13, the autofocused data does not show the same peaks at roughly the same focus times. At a 5° angle the a different sequence of events are picked than at an angle of 25° as events change position, marked by the red circles and the blue circles.

FIGURE 12) AUTOFOCUSED RESULT AT AN INCIDENCE ANGLE OF 5 DEGREES

FIGURE 13) AUTOFOCUSED RESULT AT AN INCIDENCE ANGLE OF 25 DEGREES

Due to the result when using angles from 0° to 25°. The same model was used but now 50 angles between 0° to 15° were used. The results are in Table 6. The error propagation is still present (Figure 14). As could be expected from Figure 11 when only angles up to 15° are used no sudden jumps in the reflection coefficients happen (Figure 15).
### TABLE 6) RESULTS MODEL 3 50 ANGLES OF INCIDENCE

<table>
<thead>
<tr>
<th>layer</th>
<th>input c (m/s)</th>
<th>output c (m/s)</th>
<th>rel. err. (-)</th>
<th>input $\rho$ (kg/m³)</th>
<th>output $\rho$ (kg/m³)</th>
<th>rel. err. (-)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1500</td>
<td>1500</td>
<td>0.00</td>
<td>2000</td>
<td>2000</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>2500</td>
<td>2520</td>
<td>0.01</td>
<td>2100</td>
<td>2059</td>
<td>-0.02</td>
</tr>
<tr>
<td>3</td>
<td>1600</td>
<td>1789</td>
<td>0.12</td>
<td>1900</td>
<td>1769</td>
<td>-0.07</td>
</tr>
<tr>
<td>4</td>
<td>2000</td>
<td>2138</td>
<td>0.07</td>
<td>1900</td>
<td>1831</td>
<td>-0.04</td>
</tr>
<tr>
<td>5</td>
<td>1800</td>
<td>2209</td>
<td>0.23</td>
<td>2200</td>
<td>1743</td>
<td>-0.21</td>
</tr>
<tr>
<td>6</td>
<td>2500</td>
<td>2046</td>
<td>-0.18</td>
<td>2100</td>
<td>1955</td>
<td>-0.07</td>
</tr>
<tr>
<td>7</td>
<td>1700</td>
<td>2121</td>
<td>0.25</td>
<td>2300</td>
<td>1855</td>
<td>-0.19</td>
</tr>
<tr>
<td>8</td>
<td>1950</td>
<td>2724</td>
<td>0.40</td>
<td>2000</td>
<td>1921</td>
<td>-0.04</td>
</tr>
<tr>
<td>9</td>
<td>2100</td>
<td>2013</td>
<td>-0.04</td>
<td>1900</td>
<td>1921</td>
<td>0.01</td>
</tr>
</tbody>
</table>

#### FIGURE 14) RELATIVE ERRORS DENSITY AND SEISMIC VELOCITY FOR MODEL 3
MODEL 1: 75 ANGLES AND THE EFFECT OF NOISE

To see the effect of noise on the input data a random error of between -5% and 5% was added to the amplitudes picked for model 1. The same was done again for model 1 but now with a random error of between -10% and 10%. The results are listed below (Table 7 and Table 8). The results show a clear increase in relative errors from no noise as previously described to ±5% random error and an even greater relative error when a ±10% random error is added to the amplitudes. Comparing the reflection coefficients of the first reflector in Figure 8 with the reflection coefficients, as can be seen in Figure 16 and Figure 17, show an increasing amount of fluctuation around the steady curve upwards created by the input reflection coefficients

TABLE 7) RESULTS - 5% TO 5% RANDOM ERROR

<table>
<thead>
<tr>
<th>layer</th>
<th>input c (m/s)</th>
<th>output c (m/s)</th>
<th>rel. err. (-)</th>
<th>input ρ (kg/m³)</th>
<th>output ρ (kg/m³)</th>
<th>rel. err (-)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1500</td>
<td>1500</td>
<td>0</td>
<td>2000</td>
<td>2000</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>2500</td>
<td>2591</td>
<td>0.04</td>
<td>2100</td>
<td>1987</td>
<td>-0.05</td>
</tr>
<tr>
<td>3</td>
<td>1600</td>
<td>2078</td>
<td>0.30</td>
<td>1900</td>
<td>1495</td>
<td>-0.21</td>
</tr>
<tr>
<td>4</td>
<td>2000</td>
<td>2363</td>
<td>0.18</td>
<td>1900</td>
<td>1627</td>
<td>-0.14</td>
</tr>
</tbody>
</table>
FIGURE 16) REFLECTION COEFFICIENTS FIRST REFLECTOR MODEL 1 (-5% TO 5% RANDOM ERROR)

FIGURE 17) REFLECTION COEFFICIENTS FIRST REFLECTOR MODEL 1 (-10% TO 10% RANDOM ERROR)

<table>
<thead>
<tr>
<th>layer</th>
<th>input c (m/s)</th>
<th>output c (m/s)</th>
<th>rel. err. (-)</th>
<th>input ρ (kg/m³)</th>
<th>output ρ (kg/m³)</th>
<th>rel. err. (-)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1500</td>
<td>1500</td>
<td>0.00</td>
<td>2000</td>
<td>2000</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>2500</td>
<td>2645</td>
<td>0.06</td>
<td>2100</td>
<td>1941</td>
<td>-0.08</td>
</tr>
<tr>
<td>3</td>
<td>1600</td>
<td>2323</td>
<td>0.45</td>
<td>1900</td>
<td>1316</td>
<td>-0.31</td>
</tr>
<tr>
<td>4</td>
<td>2000</td>
<td>2569</td>
<td>0.28</td>
<td>1900</td>
<td>1472</td>
<td>-0.23</td>
</tr>
</tbody>
</table>
4. Conclusions

Although the recursive scheme for calculating seismic velocity and density shows an error with each step it is not a steady trend upwards. All three models contain a layer where the relative error is larger than the one in the layer underlying it. The bottom layers do contain larger errors than the top layers. A reason for the fluctuating relative error for the densities can be that the reflection coefficients, which are found after autofocusing are either to all too high or too low. If for example two layers somewhere in the middle are taken and the first layer contains a large relative error it can still result in a smaller relative error for the bottom parameter, because the used reflection coefficient is not the true reflection coefficient.

Adding noise to the signal increases the relative error for all the parameters. The parameters in the first layer of the model will always show a relative error of zero because these are input values. The calculated values for the others layers show an increasing error as the noise on the signal goes from no deviation to 10% deviation from the initially obtained values. Increasing the noise even further than a 10% error leads to an even greater deviation from the true values of the velocity and density.

Although the maximum angle of incidence was 10° off from the critical angle for model 2 and 3 there were some unexpected results. The results show (Figure 13) that at a certain point the amplitude-picking algorithm finds eight reflectors for model 3, but the reflection events have changed order. The picking algorithm does not take this into account, so it links the wrong amplitude to the wrong reflection event. The reflection responses after autofocusing for a 5° and 25° angle of incidence clearly show this (Figure 12 and Figure 13). This results in an incorrect input in the script and thus results in large relative errors, up to 127%. Special care has to be taken to track events so amplitudes found can be linked to the correct reflector.

4.1 Acknowledgement

I want to thank Evert Slob and Jürg Hunziker for their time and help.

5. Literature


6. APPENDICES

APPENDIX A: MULTIPULSESOLUTION.m

clear all;
close all;
clc;

%% Modeling parameters
rhomod=[2000 2100 1900 1900]; % densities
velmod=[1500 2500 1600 2000]; % velocities
hmod=[175 150 140 0]; % velocity source/trigger layer
alpha=[5 20]; % angles of incidence
ref=length(rhomod)-1; % number of reflectors

%% Amplitude extraction
tic

for n=1:length(alpha);
    [rmod,Pt,t,Ptrose]=RoseVin(alpha(n),rhomod,velmod,hmod);
    %plot(t,Ptrose);
    r(:,n)=primypick(Pt,t,ref);
    rrose(:,n)=primypick(Ptrose,t,ref);
    rtrue(:,n)=(rmod');
end

%% Layer specific density and velocities (Ideal)
c=zeros(length(r(:,1)+1),1);
c(1)=velmod(1);
rho=zeros(length(r(:,1))+1,1);
rho(1)=rhomod(1);
Z=zeros(length(r(:,1))+1,length(alpha));
p=((sind(alpha)/c(1)).^2)';

% ratio test
Zratio=zeros(length(r(:,1)),length(alpha));
% used for testing errors
% err1=3; % as percentage
% r(:,1)=r(:,1)+r(:,1)*err1/100;
% err2=0;

20
\%r(:,2)=r(:,2)+r(:,2)*err2/100;

for n=1:length(r(:,1));
    
    \[ Z(n,:) = (\sqrt{\frac{1}{(c(n,:)^2)} - (p)})/(\rho(n)\^2); \]

    \%if ((1/(c(n,:)\^2))-(p)<zeros(length(p),1));
    \%fprintf('ideal: ((1/(crose(n,:)\^2))-(p))<0');
    \%break
    \%end

    Z(n+1,:)=Z(n,:)*((1-r(n,:))/(1+r(n,:)));

    A=[ones(length(alpha),1) -Z(n+1,:\^2)'];
    B=A';
    C=B*A;
    E=(B*p);
    temp=C\E;
    c(n+1)=sqrt(1/temp(1));
    rho(n+1)=sqrt(temp(2));
    Zratio(n,:)=Z(n,:)/Z(n+1,:);
end

figure
plot(alpha,Zratio(1,:),alpha,Zratio(2,:),alpha,Zratio(3,:));
xlabel('angle of incedence (degrees)');ylabel('Z ratio');
title('Z ratio');
legend('1 ref','2 ref','3 ref');

%% layer specific density and velocities (Ideal)
crose=zeros(length(r(:,1))+1),1);
crose(1)=velmod(1);
rhorose=zeros(length(r(:,1))+1),1);
rhorose(1)=rhomod(1);
Zrose=zeros(length(r(:,1))+1),length(alpha);
p=((sind(alpha)/c(1)\^2)';

Zrat=zeros(length(r(:,1)),length(alpha));

for n=1:length(r(:,1));
    Zrose(n,:)=(sqrt{((1/(crose(n,:)\^2))-(p))}/rhorose(n\^2));

    \%if ((1/(crose(n,:)\^2))-(p)<zeros(length(p),1));
    \%fprintf('((1/(crose(n,:)\^2))-(p))<0');
    \%break
    \%end

    Zrose(n+1,:)=Zrose(n,:)*((1-rrose(n,:))/(1+rrose(n,:)));
A = [ones(length(alpha), 1) (-Zrose(n+1,:).^2)'];

B = A';
C = B*A;
E = (B*p);
temp = C\E;
crose(n+1) = sqrt(1/temp(1));
rhorose(n+1) = sqrt(temp(2));
Zrrat(n,:) = Z(n,:)./Z(n+1,:);

end

figure
plot(alpha, rrose(1,:), '-'), alpha, rtrue(1,:), '-.', alpha, rrose(2,:), '-','alpha, rtrue(2,:), '-.', alpha, rrose(3,:), '-','alpha, rtrue(3,:), '-' );
axis([0 25 -0.5 0.5]);
xlabel('angle of incedence (degrees)'); ylabel('reflection coefficient');
title('reflection coefficient');
legend('1st output reflection','1st input reflection','2nd output reflection','2nd input reflection','3rd output reflection','3d input reflection');

figure
plot(alpha, Zrrat(1,:),alpha, Zrrat(2,:),alpha, Zrrat(3,:));
xlabel('angle of incedence (degrees)'); ylabel('Z ratio rose');
title('Z ratio');
legend('1 ref','2 ref','3 ref');

%error propagation
rhoerr = (rhorose-rhomod')./rhomod';
cerr = (crose-velmod')./velmod';

figure
layer=0:length(hmod)-1;
plot(layer, rhoerr, '-', layer, cerr, '--')
xlabel('layer'); ylabel('relative error');
legend('rel. dens. error','rel. vel. error');
toc

APPENDIX B: ROSEVIN.M

function [rmod,Pt,t,Ptrose]=RoseVin(alpha,rhomod,velmod,hmod)
%clear all
%close all
%clc

rho=rhomod;       % densities
vel=velmod;       % velocities
h=hmod;
nl=length(h);
p = sind(alpha)/vel(1);
slo = sqrt(vel.^(-2)-p.^2);
kap=1./(rho.*slo.^(-2));
Z=sqrt(kap./rho);
r=(Z(1:nl-1)-Z(2:nl))./(Z(1:nl-1)+Z(2:nl));
r(nl)=0;
x3=linspace(0,2e3,nx);
dz=x3(2);
t3=zeros(1,nx);
impx(1,1:nx)=1/sqrt(Z(1));
nh=round(h/dz);
t3(1:nh(1))=(1:nh(1))^dz/vel(1);

for il=2:nl-1
    t3(sum(nh(1:il-1))+1:sum(nh(1:il)))=t3(sum(nh(1:il-1)))+ ... 
        (1:nh(il))^dz/vel(il);
    impx(sum(nh(1:il-1))+1:sum(nh(1:il)))=1/sqrt(Z(il));
end

t3(sum(nh(1:il))+1:nx)=t3(sum(nh(1:il)))+(1:nx-sum(nh(1:il)))*dz/vel(nl);
impx(sum(nh(1:il))+1:nx)=1/sqrt(Z(nl));
cf=60;
nf=16*8192;
freq=linspace(0,800,nf);
df=freq(2)-freq(1);
freq=freq-1i*df;
cf=60;
wav=2*sqrt(1/pi)*(freq/cf).^2.*exp(-(freq/cf).^2)/cf;
R0=zeros(1,nf);
P=zeros(1,nf);
for ii=nl-1:-1:1
    gam=2i*pi*freq/vel(ii+1);
    R0=(r(ii)+R0.*exp(-2*gam*h(ii+1)))./(1+r(ii)*R0.*exp(-2*gam*h(ii+1)));
    P=(r(ii+1)+P).*exp(-gam*h(ii+1));
end
gam=2i*pi*freq/vel(1);
R0t=2*real(ifft(R0.*exp(-2*gam*h(1)),2*nf));
P=(r(1)+P).*exp(-gam*h(1));
Pt=2*real(ifft(P.*wav,2*nf))*df*2*nf;
t=(0:nf-1)/(2*nf*df);
dt=t(2)-t(1);
Pt=Pt(1:nf).*exp(2*pi*t*df);
R0t=R0t(1:nf).*exp(2*pi*t*df);

figure(3)
plot(t,R0t(1:nf))
axis([0 4 -0.1 0.32])
kids=get(gca,'children');
set(kids,'linewidth',2)
set(gca,'fontsize',18)
xlabel('two-way travel time (s)')
ylabel('reflection response')
title('convolved with Gaussian')

nfc=nf/128+nf/256;
nf=2*nfc;
k=zeros(nfc,nf);
ds=1/(2*nf*dt);
s=2i*pi*fftshift(-nt:nt-1)*ds;
tic
img=zeros(1,nfc);
for tf=1:nfc
    Rc=zeros(1,2*nt);
    Rc(1:nt+nfc)=R0t(1:nt+nfc);
    R0f=fft(Rc);
    w0=Rc;
    w0=circshift(w0,[0,-tf+1]);
    w0(tf:2*nt-tf+1)=0;
    diff=1;
    ii=0;
    while diff > 1e-3
        ii=ii+1;
        if floor(ii/30)==ii/30
            disp(['ii= ',num2str(ii,4)])
        end
        w1=real(ifft(R0f.*(exp(s*(tf-1)*dt)-fft(circshift(fliplr(w0),[0 1])))));
        w1(tf:2*nt-tf+1)=0;
        diff=sqrt(sum(abs(w0-w1).^2)/sum(abs(w0).^2));
        w0=w1;
    end
    tmp=fftshift(w0);
    k(tf,:)=tmp(nt-nfc+1:nt+nfc);
end
df=1/(2*nfc*dt);
freq=df*fftshift(-nfc:nfc-1);
f0=60;
wav=2*sqrt(1/pi)*(freq/fc).^2.*exp(2i*pi*freq*dt/(2*pi)-((freq/fc).^2)/fc);
pimg=2*real(ifft(wav.*fimg,2*nfc))*2*nfc*df;
figure(2)
plot(t(1:nfc),Pt(1:nfc),'b',t(1:nfc),pimg(1:nfc),'r--')
set(gca,'fontsize',18)
xlabel('focus time (s)')
ylabel('signal strength (-)')
legend('model reflections','autofocus result')
end
rmod=r;
t=t(1:nfc);
Pt=Pt(1:nfc);
Ptrose=pimg(1:nfc);

end

APPENDIX C: PRIMYPICK.M

function [r,temp]=primypick(trace,t,ref)
%% spanning dimensions
l=zeros(ref,2);
r=zeros(ref,1);
n=1;
i=1;

%% Finding signal
for i=1:ref;
    while abs(trace(n))<=0.01;
        n=n+1;
    end
    l(i,1)=n;

    while abs(trace(n))>0.01 || abs(trace(n+1))>0.01;
        n=n+1;
    end
    l(i,2)=n;
    ltrace=length(trace(l(i,1)-1:l(i,2)+1));
    temp=nextpow2(ltrace);
    temp=2^(temp);
    temp=zeros(1, temp-ltrace);
    temp=[trace(l(i,1)-1:l(i,2)+1) temp];
    [traceint dt]=sincinterpol((t(2)-t(1)),temp,10);
    xsize2 = length(traceint); xvec2 = (0:xsize2-1)*dt+t((l(i,1)-1));
    %figure;
    %plot(trace(l(i,1)-1:l(i,2)+1),t(l(i,1)-1:l(i,2)+1),'b'); hold on
    %plot(traceint,xvec2,'r'); hold off
    %grid on; legend('original','interpolated');title(i);
    r(i)=real(max(traceint));
    %if r(i)== max(trace(l(i,1):l(i,2)))
    %    r(i)=-real(r(i));
    %else
    %end

%r(i)=real(r(i));
%end
end

APPENDIX D: SINCINTERPOL.M

function [fxout dx2] = sincinterpol(dx,fxin,nfac)
% Applies Sinc-Interpolation to vector.
% USAGE: [fxout dx2] = sincinterpol(dx,fxin,nfac)
%
% INPUT:
% dx: the spacing of the coordinates
% fxin: the function to be interpolated
% nfac: the factor that specifies how many more points there will be in
%       the interpolated function
%
% OUTPUT:
% fxout: interpolated function
% dx2: new spacing of the coordinates
%
% EXAMPLE:
% clear all; close all; clc;
% nfac = 64; dx = pi/8; xsize = 16;
% fs = 18; % Fontsize
% xvec = (0:xsize-1)*dx; fx = sin(xvec);
% [fx2 dx2] = sincinterpol(dx,fx,nfac);
% xsize2 = length(fx2); xvec2 = (0:xsize2-1)*dx2;
% figure;
% plot(xvec,fx,'b'); hold on
% plot(xvec2,fx2,'r'); hold off
% grid on; legend('original','interpolated')
% set(gca,'Fontsize',fs)

    xsize = length(fxin);
dx2 = dx/nfac;
dkx = 2*pi/(xsize*dx);
temp = (xsize)*ifft(fftshift(fxin))*dx;
temp2 = zeros(1,xsize*nfac);
temp2(1:length(temp)/2) = temp(1:length(temp)/2);
temp2(xsize*nfac-length(temp)/2+1:xsize*nfac) = temp(length(temp)/2+1:length(temp));
fxout = fftshift(fft(temp2))*dkx/(2*pi);