NTNU – Trondheim
Norwegian University of
Science and Technology

Global Optimization and Inital Models In Seismic Pre-Stack Inversion

Øyvind Aunan Øvstegård

Earth Sciences and Petroleum Engineering
Submission date: June 2012
Supervisor: Egil Tjåland, IPT

Norwegian University of Science and Technology
Department of Petroleum Engineering and Applied Geophysics

Preface

This thesis is submitted in fulfillment of the requirement for the degree of M.Sc. at the Norwegian University of Science and technology (NTNU). The thesis work has been done at the department of Petroleum Engineering and Applied Geophysics. I thank Egil Tjåland for being my advisor, and assisting me with this thesis. I thank the students at room number 100 at the department of Petroleum Engineering and Applied Geophysics, NTNU for interesting and helpful discussions.

Trondheim, June 2012.

Øyvind Aunan Øvstegård

Abstract

Pre stack inversion of seismic data consists of numerous difficulties. Two of the problems of greatest concern are the problems of non-uniqueness and non-linearity of the inversion. There may exist several solutions to any given inversion problem, and to be able to choose the correct solution we are dependent on a priori information. This thesis will explain how a priori information can be implemented with the seismic data using Bayesian modeling and fractal based initial methods in order to obtain the most likely solution for the inversion.

This thesis will also explain the theory behind global optimization routines, such as the random walk Monte Carlo, the Metropolis algorithm and Simulated Annealing. A Simulated Annealing routine has been made, and this is used to solve optimization problems. The routine is analyzed for its capability of finding global optimums and the requirements for its success. It is then implemented to simulate the inversion of a seismic dataset. The solutions of the inverted data is then analyzed and compared to the actual solution. This is done for an uncontaminated dataset, and for a dataset containing noise.

The work has shown that Simulated Annealing can be a good method for finding a global optimum, but that the global optimization routine is unable to produce good results without good constraints and a good initial model, due to the problem of non-uniqueness.

Sammendrag

Pre stack inversjon av seismiske data byr på en del problemer. To av de største problemene er det med de ikke-unike løsningene og det ikke-linære forholdet mellom de modellerte og de faktiske seismiske dataene. Det kan eksistere flere løsninger til et gitt inversjonsproblem, og for å velge den riktige løsningen er vi avhengige av tilleggsinformasjon om våre data. Denne oppgaven vil forklare hvordan denne tilleggsinformasjon kan implementeres med de seismiske dataene ved hjelp av Bayesisk modellering og fraktalbaserte inisielle modellere for å oppnå den mest sannsynlige løsningen. Denne oppgaven vil også forklare teorien bak globale optimalisering rutiner, slik som random walk Monte Carlo, Metropolis algoritmen og Simulated Annealing. En Simmulated Annealing rutine har blitt lagd, og denne vil brukes for å løse optimaliseringsproblemer.

Simulated Annealing rutinen er analysert for sin evne til å finne globale optimum og hvordan parameterinnstillingene må være for at den best skal klare det.

Løsningene av de inverterte data blir deretter analysert og sammenlignet med den faktiske løsningen. Både rene data og data som har blitt tillagt støy blir invertert. Dette for å vise konsekvensene støy vil gi inversjonen. Resultatene viser at Simmulated Annealing kan være en veldig effektiv global optimaliseringsmetode, men at den globale løsningen ikke nødvendigvis er den rette løsningen når dataene inneholder støy. Den inisielle modellen for de inverterte parameterene er derfor veldig vikitg, og er med på å farge inversjonen

Table of Content

Table of figures

1 Introduction

Seismic inversion can generally be described as the determination of subsurface properties from seismic data. Seismic data can be described as an interface property, while seismic inverted data can be described as a rock property. There are several types of inversion. Inversion can be done both post-stack and pre-stack. The data can be inverted in a deterministic or a stochastic matter. It can be model based or recursive. The problem with most model based inversion schemes is that the relationship between the model and the real seismic data is assumed to be linear. In fact, this is very seldom the case. If this is assumed we require that the global minimum is in the neighborhood of our initial guess in order to find it. If our initial guess is wrong, we will not be able to find the global minimum.

This thesis will focus on pre-stack seismic inversion using Monte Carlo methods in general and Simulated Annealing specially. These are global optimization techniques which choose parameters randomly and calculate their fitness to a certain inversion problem. By doing this we hope that our solution will not be a local minimum, but that we find the global minimum, which should be the best possible solution. The solution should be found, regardless of initial choice of parameters.

The difficulties in seismic inversion with respect to implementation of a priori information together with a global optimization routine will be discussed. It will be done by going through the theory behind Bayesian modeling and fractal based initial models. The global optimization theory behind Simulated Annealing will be explained, together with its origin from Markov Chain Monte Carlo and the Metropolis algorithm.

The Simulated Annealing method is regarded as an effective method of finding a global minimum in a non-linear inversion problem (Rothman, 1985). The thesis will show examples of a Simulated Annealing algorithm used to find global minima in both basic optimization problems and a simulation of a synthetic seismic inversion problem. This will be done by a simple inversion routine showing the principles of how Simulated Annealing can be used in a full scale inversion of a seismic dataset. It will also discuss the drawbacks and difficulties of this method, and address the important factors in getting the best possible inversion result.

2 Pre stack inversion and the inversion parameters

The main reason to conduct a seismic pre-stack inversion is to be able to get as much information as possible from the seismic data. The reflection coefficients contain information about the elastic parameters of the subsurface. By finding these parameters we may be able to attach physical properties to the seismic data, thereby increase our understanding of the subsurface.

Properties that can be found using seismic inversion are considered to be (Dahl, 1990):

- Seismic velocities
- Densities
- Elastic parameters
- Impedances
- Zero offset reflection coefficients
- Attenuation factors
- Parameters describing the geometry of layers

The reason we can say something about these parameters is that when a plane pressure wave (P-wave) or a vertical polarized shear wave (SV-wave) meets an interface between two layers at a non-vertical angle, four plane waves are created. These are reflected P- and SV-waves and transmitted P- and SV-waves. This means that the reflection coefficient matrix can be written in the following form (Tarantola, 1987):

$$R = \begin{bmatrix} R_{pp} & R_{sp} \\ R_{ps} & R_{ss} \end{bmatrix} \tag{2.1}$$

Where the first subscript represents the down-going wave and the second represent the up-going wave, where p is a pressure wave and s is a shear wave.

Seismic data can be inverted by only regarding R_{pp} or by regarding both R_{pp} and R_{ps} , so called joint inversion. An exact expression for the entire reflection matrix can be found in (Zoeppritz, 1919). The problem with the Zoeppritz equations is that they are regarded to be highly non-linear, and rely on six parameters for any given angle. These are the P-wave velocity, the S-wave velocity and the density on both sides of the reflecting interface. However, only four of these parameters are independent.

There has been made a lot of equations which are good approximation of the Zoeppritz equations when contrast between layers and incident angles are small. Aki and Richard (1980) or Smith and Gidlow (1987) are examples of these. These are linear approximations of the Zoeppritz equations, and are favorable because of the reduction of unknown parameters.

In these approximations the PP reflection coefficient relies on only four parameters. This may for example be the ratio between the two densities, the ratio between the two P-wave velocities and the ratio between the P- and S-wave velocities in the two separate layers. This means that if densities or velocities are changed by the same factor, the reflection amplitude will not change. However, if we use pre critical PP data we are only able to establish three of these parameters (Ursin, 1996)

Since the common used equations that represent seismic reflection coefficients are defined as a function of angle, we also want to have our seismic data as a function of angle. A seismic Common Mid-Point (CMP) gather can be regarded as the amplitude variations with angle, when all other amplitude effects are accounted for and Normal Move Out (NMO) is conducted. It is therefore important that this is done before the inversion is conducted.

2.2 The Forward Problem

In this paper the linear approximation of the Zoeppritz equation rearranged by Fatti et al. (1994) has been used. This is a total representation of an elastic earth (Ma, 2002):

$$R_{pp}\,(\theta) = [1 + tan^2\theta]R_p - \left[8\left(\frac{V_s}{V_p}\right)^2 sin^2\theta\right]R_s - \frac{1}{2}\left[tan^2\theta - 2\left(\frac{V_s}{V_p}\right)^2 sin^2\theta\right] \quad (2.2)$$

where:

$$R_p = \frac{1}{2}[\frac{\Delta V_p}{V_p} + \frac{\Delta\rho}{\rho}] \qquad R_s = \frac{1}{2}\left[\frac{\Delta V_s}{V_s} + \frac{\Delta\rho}{\rho}\right] \qquad R_d = \frac{\Delta\rho}{\rho} \qquad (2.3)$$

For small reflectivity series we can use the approximations:

$$L_p(i) = \frac{1}{2}[\ln Z_p(i+1) - \ln Z_p(i)] \qquad (2.4)$$

$$L_s(i) = \frac{1}{2}[\ln Z_s(i+1) - \ln Z_s(i)] \quad , \qquad (2.5)$$

where V_p and V_s are the average acoustic and shear velocities over the two interfaces, respectively. θ is the angle of incident of the reflected wave. Z_p and Z_s is the average acoustic and shear impedances over the two layers, respectively. ρ is the average density over the two layers, and $\Delta\rho$ is the density difference between the lowermost and uppermost layer.

Equations (2.4) and (2.5) represent the logarithmic differences of P- and S-impedance values in the two layers. This is a fair assumption if the reflectivity series are under 0.5. If we substitute equation (2.4) and (2.5) into equation (2.2) we obtain equation (2.6).

$$R_{pp}(\theta) = [1 + tan^2\theta]L_p - \left[8\left(\frac{Z_s}{Z_p}\right)^2 sin^2\theta\right] L_s \; - \frac{1}{2}\left[tan^2\theta - 2\left(\frac{Z_s}{Z_p}\right)^2 sin^2\theta\right] R_d \qquad (2.6)$$

In equation (2.6) we have also used the relationship that $Z_p = V_p\rho$ and $Z_s = V_s\rho$

With these equations the number of unknown parameters is reduced to three, the average acoustic impedance, the average shear impedance and the relative change in density over the two layers. This is an approximation of the Zoeppritz equation and is only valid for small angles and small variations in properties over the interface. Further, this representation assumes a horizontally layered earth model and the two media to be isotropic with a welded contact (Ma, 2002). As we can see from equation (2.6) the reflection coefficients does not change if the acoustic and shear impedance changes by the same factor. These reflection coefficients are what we in theory obtain when the seismic data is processed and all other amplitude effects than the reflections are removed.

Model based inversion methods use the forward model to create a synthetic response. The relationship between the synthetic and the real seismic data can be described as:

$$d_0 = d(m) + n \qquad (2.7)$$

Where m is the model parameters, d_0 is the observed seismic data, $d(m)$ is the modeled synthetic data, and n is the noise.

It is the difference between the seismic data d_0 and the modeled data $d(m)$ we want to minimize by varying the model parameters m. By doing this we are able to find the rock properties m which gave the seismic response d_0.

As we can see from equation (2.7) we will not get exactly the same response from the model as the real seismic data. The reasons for this are:

- The real seismic data contains noise
- The mathematical description of wave propagation in the earth is not correct.
- Simplifying assumptions about the earth parameters are made. The equations for reflection coefficients are approximate.

We can also see from equation (2.7) that the relationship between the model parameters m, and the synthetic seismic data d(m) may be strictly non-linear. If these strictly non-linear functions are optimized linearly we are highly dependent on the initial estimates of the model parameters in order to find the global minimum. This is because when a function has several local minima, the function can't be solved using a set of linear equations. No such set of linear equations exist unless other assumptions are made.

A complete search of the model space are in many cases not computational feasible because of the amount of possible solutions. For example if there are a parameters, where each parameter can have b values, the number of possible solutions will be b^a. With a couple of unknown parameters the number of possible solution will quickly become enormous.

2.2 General Pitfalls in Seismic Inversion

There are several factors which affect our inversion algorithm which needs to be accounted for. These are:

2.1.1 Non-Uniqueness

There may be several solutions which all explain the model equally well, even though the model parameters vary extensively. If this is the case there is no unique solution to our inversion problem.

This may happen if several solutions have zero prediction. Since the seismic data is limited in bandwidth both by an upper and lower boundary we do not have all the information about the properties of the reflecting interface as we may need to have one unique solution with zero prediction error. This means that our model may be a good fit to our seismic band limited data but may not have been a good fit if the lower and upper areas where we do not have any information where known. We may also experience that none of the solutions have zero prediction error, but several solution has the same low prediction error. This means that none of the solutions fit the model perfectly but several solutions have the same close estimate.

2.1.2 Existence

In order to invert the seismic data correctly there need to exist a solution to the inverse problem. An example where an analytic solution does not exist is if we have more unknowns than equations. This is called an underdetermined problem. This happens when the solution does not provide the information necessary to determine the model parameters uniquely. To resolve this problem extra information need to be added to single out one of the many possible solutions. Such information is called *a priori* information. This information can for example state that the unknown parameter must be in a certain range. This may greatly reduce the number of possible solutions and help us choose the correct solution. The problem with a priori information is that it may in itself be an uncertainty and we need to evaluate the probability of these constrains in our model.

2.1.3 Stability

Stability is a measurement of how the model responds to small errors. If our model shows instability it will change drastically to many small errors. Because of presence of noise in our dataset such small errors commonly occur and it is therefore important that this does not affect our inversion considerably. An unstable inversion may therefore alter our model completely, giving a wrong inversion and create non-uniqueness.

2.1.4 Robustness

Robustness is how the model replies to few large errors. Large errors may come from different types of noise. It is therefore important that our model is robust so these large errors do not affect our model considerably. Lack of robustness may also create non-uniqueness.

2.1.5 Inversion intervals

Noise is generally considered to be larger at the bandwidth limits in the seismic data. Close to the upper frequency boundary, noise can cause a significantly greater influence than the data itself. This limitation applies also to our decision of how many intervals or layers we should invert. If we choose too many reflections close together it will cause noise to be modeled as reflections resulting in a greater uncertainty in the inverted acoustic impedances. This is called “over fitting” the data.

2.1.6 Absolute impedance

There may also be some limitations in the lower boundary of the frequency spectrum. The lower frequencies contain the information about the absolute values of the acoustic impedance. This means that it is impossible to recover the absolute acoustic impedances from a seismic trace. The inversion can only give us the relative change in impedance. To get the absolute impedances from the relative we therefore need to add the low frequencies from well data. This is often done by an interpolation over the frequencies and impedances from the well data. This problem may also create non-uniqueness.

3 Monte Carlo Methods

A problem arising in most inversions procedure is that the relationship between the model and the real data is often assumed to be linear and a linear approach is therefore conducted. If the initial model is chosen nearby a local minimum we will not get the optimal solution which should be the global minimum.

To solve non-linear inversion problems we may therefore use a Monte Carlo method which may create an acceptable solution in a shorter time period than the complete exploration of the model space, and not be highly dependent on the initial guess as a linear method would. A standard Monte Carlo inversion can be described by the following steps (Tarantola, 1987):

- Define a domain of possible inputs
- Generate random inputs by using a probability distribution
- Compute the inputs
- Reject or accept

It is based on completely random picking of samples hoping that we will get a good fit in a small number of trials. In theory it means that if a problem has a search space of N, we may find the solution we are looking for after between 1 and N searches. This has the immediate disadvantage that the routine may be very computational expensive. The advantage of Monte Carlo simulation is that it does not rely on a linear relation between the real seismic data and the synthetic seismic. The synthetic data is created in a random fashion and they are then compared to the real seismic data and the model are either rejected or accepted. To make sure that the routine does not accept un-probable solutions we can make constraints on the algorithm, for example that density or velocity can’t be negative.

Because of the problems of non-uniqueness the accepted models should be interpreted to see if the inverted parameters seem unlikely or are in great contrast with what is to be expected even though they produce a small misfit error. When the seismic data contains considerable amount of noise, linear method may fail to produce accurate results because of the then increased non-linear relationship between the initial model and the seismic data (Stoffa, 1995).

If we should use the non-linear Monte Carlo simulation or a linear inversion method is therefore highly dependent on the nature of the data-model relationship. There are examples where linearized inversion methods may be favored. For example if the data is known to contain little noise and the model parameters range are known a priori with high degree of certainty. However, as the data become more complex the error estimation in the linearized methods will also increase in complexity. This may favor Monte Carlo methods.

3.1 Markov Chain Theory

A Monte Carlo algorithm which is frequently used is the Markov Chain Monte Carlo. The reason for this algorithms popularity is that if the number of iterations is sufficiently, the chain will reach a stationary distribution, independent of the initial distribution. The requirement for this to happen is that the chain is irreducible and aperiodic. (Walsh, 2004)

A Markov chain can be described as a system which undergoes a random transition from one state to another. The transition is without memory, meaning that the next state depends only on the current state, and not the previous. This can be described for a random 1 dimensional sequence in the time domain as:

$$X = [X_{0,}X_1 \ldots.. \ X_n] \tag{3.1}$$

$$P(\, X_t = x_t | X_{t-1} = \ x_{t-1}, \ldots\ldots, X_0 = \ x_0) \ = P(\, X_t = x_t | X_{t-1} = x_{t-1}) \tag{3.2}$$

Meaning that the next time state t is only dependent on the previous time

state t-1. This means that knowledge of the history of the sequence does not give you any new information.

When we are dealing with multiple dimensions a Markov Chain is referred to as a Markov random field. If we define the random variable in two dimensions as:

$$X = \ X_{i,j} \tag{3.3}$$

Then the random variable depends on the neighborhood. The neighborhood can be defined as

$$A_{i,j} = \{(i+1,j),(i-1),j),(i,j+1),(i,j-1)\} \tag{3.4}$$

Then we can state a 2-D Markov field as:

$$P\big[X_{i,j} = \ x_{i,j}\big|X_{k,l} = \ x_{k,l}\,,(k,l) \neq (i,j)] = P[X_{i,j} = x_{i,j}|\, X_{k,l} = \ x_{k,l},(k,l) \ \in \ A_{i,j}] \tag{3.5}$$

Here $A_{i,j}$ can in theory be an arbitrary neighborhood function and does not need to be on the form of (3.4). The probability of X if also required to always be positive for all x. (Rothman, 1985)

The advantage of the Markov chain is that even though the process is generating samples randomly, we can still say something about its statistical properties. The Markov chain property is that it exhibits a Gibbs probability distribution and all Gibbs distributions define a Markov random field.

3.2 Gibbs distribution

The Gibbs distribution is a distribution based on the principles of canonical ensemble. The canonical ensemble can be described as representing the probability of microstates in a certain system. The distribution can be described as the probability *p* of finding the system in a state with a certain energy level *E*, with the constraint that the total energy on the total system is remained constant, the particle number N and the volume V is kept constant.

The Gibbs distribution can be described by the following analogy of a heat bath:

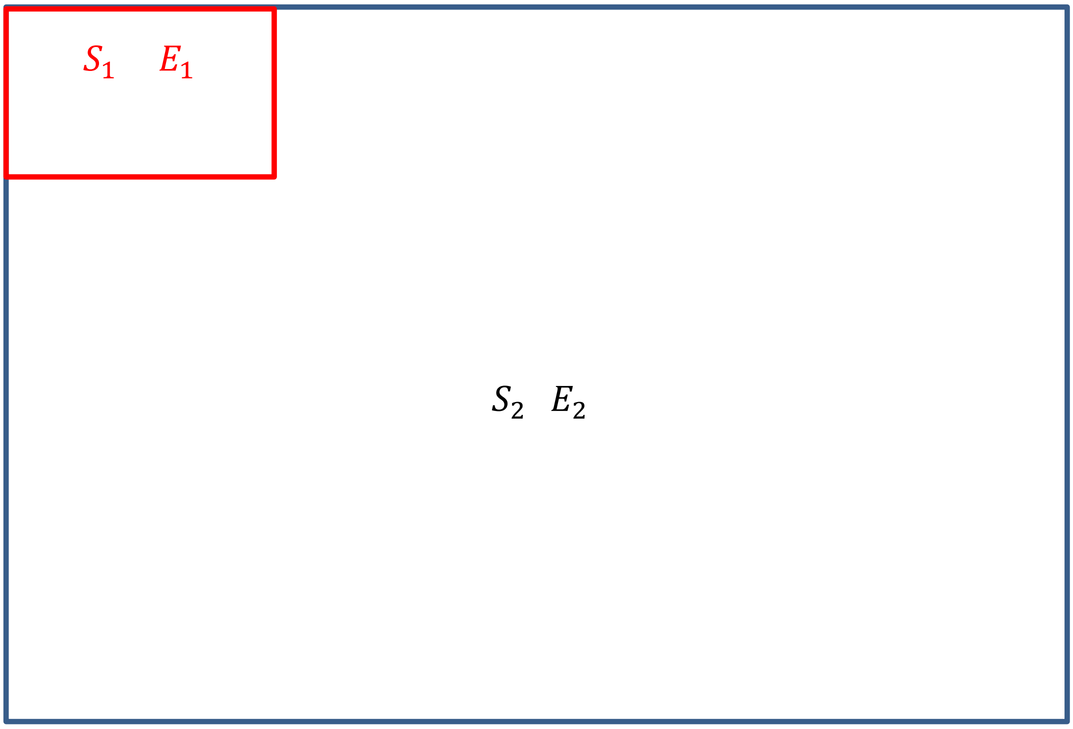

Figure 1: Illustration of a heath bath

We assume that S is a closed system. Inside S we have S_1 and S_2. S_1 is the area of interest while S_2 is the environment. $S_1 << S_2$. The energy in these areas are respectively E_1 and E_2 and E is the total energy of the system. $E = E_1 + E_2$. The total energy of the system is fixed, but E_1 and E_2 are not. The particle number of $S_1$and S_2 are fixed and energy is the only thing exchanged from the two systems.

We want to find the probability that the system S_1 is in a given state. E_1'

The total energy can then be written as $E = E_1' + E_2$.

If we assume that P_1' is the probability of S_1 being in the energy state E_1' and P_1 for S_1 being in the energy state E_1 we can then write the ratios of probabilities as:

$$\frac{P_1}{P_1'} = \frac{\Omega(E_2)}{\Omega(E_2')} = \frac{\Omega(E-E_1)}{\Omega(E-E_1')} \qquad (3.6)$$

Where $\Omega(E_2)$ defines the multiplicity when the system is a fixed value. This assumes that every state is equally likely.

By using the Gibbs entropy formula we can express this in forms of entropy:

$$S_2 = \ kln(\Omega) \ , \qquad (3.7)$$

this can be rewritten as:

$$\Omega = e^{\frac{S_2}{k}} \qquad (3.8)$$

By substituting equation (3.8) into equation (3.6) we obtain:

$$\frac{P_1}{P_1'} = \ e^{\frac{1}{k}[S_2(E-E_1) - S_2\left(E-E_1'\right)]} = \ e^{\frac{1}{k}[-(E_1-E_1')\frac{\delta S_2}{\delta E}]} = \ e^{\frac{1}{k}[-(\frac{E_1-E_1'}{T})]} = \frac{e^{\frac{-E_1}{kT}}}{e^{\frac{-E_1'}{kT}}} \qquad (3.9)$$

The second last term in equation (3.9) is found by the two term Taylor expansion. This is done by assuming that $E \gg E_1$

The definition of the second part of the second law of thermodynamics can be written:

$$\frac{\delta S_2}{\delta E} = \frac{1}{T} \quad , \qquad (3.10)$$

where T is defined as the temperature.

Finally, the probability of one given state can then be written as:

$$P_n = \ Ce^{\frac{-E_n}{kT}} \ , \qquad (3.11)$$

which is the definition of the Gibbs distribution. It can be showed that the constant C can be defined as $\frac{1}{Z}$ where Z is defined as:

$$Z = \sum_x \exp\left[\frac{-E(x)}{k_b T}\right] \tag{3.12}$$

where k_b is the Boltzmann constant. The Boltzmann constant relates energy at individual particle level with temperature. This nature of the Gibbs distribution makes it an ideal distribution in simulated annealing routines (Javanainen, 2009).

3.3 Bayesian Modeling

Because of many of the issues described in section 2.2 it is clear that inverting seismic data is not a simple procedure. As we can see, one of the biggest issues is the non-uniqueness of the inversion problem. There are a lot of different factors which may create this, and this complicates the inversion considerably.

If our inversion is a linear inversion routine the initial model becomes very important in the inversion scheme. By choosing a wrong initial model our routine may not converge to the correct solution and our inversion becomes inaccurate or wrong. To be able to get a good estimate of an initial model as possible we can formulate a prior probability distribution. By doing this we can assign low probability to unlikely areas of the search space thereby ruling them out as acceptable solutions. By doing this we ensure that the routine don't find a minimum which in reality is of low probability even though the error estimate is low.

To incorporate a priori information into the seismic inversion Bayesian modeling may be used. This can be regarded as a good method of incorporating new information into an existing model. Bayes' theorem can be given as:

$$P(M \mid D) = \frac{P(D \mid M)P(M)}{P(D)} \tag{3.13}$$

Here M is the model and D is the data, and P is the probability distribution.

The $P(M \mid D)$ can be regarded as the distribution of the model parameters given our data. The term $P(D \mid M)$ is the likelihood function and corresponds to the forward model.

$P(M)$ is the prior information we have about the model before the data is known. This may be theoretical knowledge such as that the densities can't be negative, and information gathered from the geological setting. The $P(D)$ can be regarded as the prior marginal distribution of D. When the data is observed this is just a normalizing term, and the denominator in equation (2.8) can be skipped and write the equation as a proportionality (Rabben, 2009).

Bayes theorem can be used in seismic inversion by first assuming prior distributions, variance and covariance to our model parameters. If we assume multivariate Gaussian distribution for the model parameters and the noise we may write:

$$P(M) = \mathrm{N}(M; \mu_m, \Sigma_m) \tag{3.14}$$

$$P(e) = \mathrm{N}(e; 0, \Sigma_e) \tag{3.15}$$

where the noise has an expected value of zero and μ_m is the expected value of the model parameters. Σ_m is the covariance of the prior model parameters and Σ_e is the covariance of the noise. The likelihood function can then be written:

$$P(D|M) = \mathrm{N}(D; f(M), \Sigma_e) \tag{3.16}$$

where $f(M)$ is the deterministic non-linear forward model. The covariance of the noise and data are then important in deciding if the prior information or the data has the greatest influence on the posterior distribution. If the variance of the noise is small compared to the variance of the prior information, the posterior will be mainly determined by the data, and it will be concentrated around the prior mean μ_m if the opposite is the case (Rabben, 2009).

The properties of the covariance are therefore very important for the inversion. The covariances can be defined as:

$$\Sigma_m = \sigma_m^2 C_m \tag{3.17}$$

$$\Sigma_e = \sigma_e^2 C_e \tag{3.18}$$

By introducing equation (3.17) and (3.18) we partly let the covariance be decided by the data, since the estimation of the variances σ_m^2 and σ_e^2 are part of the inversion routine and C_m and C_e are scaling factors given a priori. Then we can write the new likelihood distribution on the following form:

$$P(m|\sigma_m^2) = \mathrm{N}(M; \mu_m, \sigma_m^2 C_m) \tag{3.19}$$

$$P(d|m, \sigma_m^2) = \mathrm{N}(D; f(m), \sigma_e^2 C_e) \tag{3.20}$$

with these equations we can write the total posterior distribution in Bayes theorem as:

$$P(m, \sigma_m^2, \sigma_e^2|d) \propto P(d|m, \sigma_m^2, \sigma_e^2) P(m, \sigma_m^2, \sigma_e^2) = \\ P(d|m, \sigma_e^2) P(m|\sigma_m^2) P(\sigma_m^2) P(\sigma_e^2) \tag{3.21}$$

This means that we can calculate the probability of a model based on our prior knowledge and information about the data. Then we can constantly update our model when new information is available and compare different models to each other based on the probability of their correctness. As equation (3.21) shows we now have an expression for the probability distribution for the different model parameters, their variance and the noise. The variance of the noise and the prior information can also be implemented on such a way, that the information we see as most likely is given the most influence. Bayesian modeling may help us implement as much information as possible, it is however important to note that a Bayesian approach does not remove the problems of non-uniqueness, it only helps identifying it (Rabben, 2009).

3.4 Bayesian Modeling and The Gibbs distribution

As mentioned earlier it is of paramount importance to incorporate a priori information into our probability distribution. It can be showed that the posterior probability, i.e. the conditional probability after all relevant evidence is taken into account, P(X=x | D = d) is also a Gibbs distribution. This can be showed by using Bayes theorem (2.8) If we substitute Gibbs prior for P(X=x) and keep P(d=d) a constant we obtain:

$$P(X = x|D = d) = \frac{1}{Z} P(D = d|X = x) \exp\left[\frac{-E(x)}{T}\right] \qquad (3.22)$$

Z is now a new constant because it is a combination with the constant in equation (2.8).

We then include noise into the model. If we assume the noise to be Gaussian or exponential, identically distributed, independent of X and with a zero mean.

The noise can then be defines as:

$$P(N = n) = \frac{1}{c} \exp[-\frac{1}{2}(\frac{\|n\|_{p}{}^{p}}{\sigma})] \qquad (3.23)$$

Where c and are constant and $\|n\|_{p}$is the L_p norm. If p = 1 the noise is exponential and if p = 2 it is Gaussian.

The posterior can then be written:

$$P(X = x|D = d) = \frac{1}{Z} P(D = G(x) + n|X = x) \exp\left[\frac{-E(x)}{T}\right] =$$

$$P(X = x|D = d) = \frac{1}{Z} P(N = d - G(x)|X = x) \exp\left[\frac{-E(x)}{T}\right] \quad (3.24)$$

Since the Noise N is independent of X:

$$P(X = x|D = d) = \frac{1}{Z} P(D = d|X = x) \exp\left[\frac{-E(x)}{T}\right] \quad (3.25)$$

By substituting in equation number (3.14) we obtain:

$$P(X = x|D = d) = \frac{1}{Z} \exp\left[\frac{-E(x)}{T} - \frac{1}{2}\left(\frac{||n||_p{}^p}{\sigma}\right)\right] \quad (3.26)$$

If we then write:

$$E(x,d) = E(x) + \frac{T}{2}\left(\frac{||d-G(x)||_p}{\sigma}\right)^p \quad (3.27)$$

Then:

$$P(X = x|D = d) = \frac{1}{Z} \exp\left[\frac{-E(x,d)}{T}\right] \quad (3.28)$$

And it is shown that the posterior distribution is also a Gibbs distribution (Rothman, 1985).

3.5 Metropolis algorithm

The Metropolis algorithm is a Monte Carlo routine which can be described in the following way: An initial model is chosen according to an educated guess based on a priori information. The energy for this model is calculated. In our case the energy can be analogous to for example the least square or the absolute error between our synthetic seismic and the real seismic data. The initial parameters are then used as basis to create a new random guess. This can be done from a probability distribution or from a random perturbation. Then the new energy is calculated. If this energy is lower than the original one, this energy is accepted as the new model. If the energy is higher it is accepted with a probability of

$$P(\Delta E) = e^{\frac{\Delta E}{T}} \tag{3.29}$$

Where ΔE is the change in energy state and T is the temperature. As shown in the last section, equation (3.29) is the Gibbs distribution.

The criteria in (3.29) is easily fulfilled by choosing a random number α from an uniform distribution between 0 and 1 and accept the new energy if

$$\alpha < P(\Delta E) \tag{3.30}$$

If this is not the case, the existing values are retained. This is called the Metropolis criteria. Because of this criterion the algorithm has the possibility to make uphill moves. In this way it is possible for the algorithm to “climb” out of a local minimum.

3.6 Fractal based initial models

As we can see from the Bayesian Modeling section it may be important to choose the initial model carefully in order to get an acceptable solution. This may for example apply in circumstances where the variance of the noise is large, and the model is more influenced by the initial guess. If the initial model is completely wrong in these circumstances the inversion will also fail to give reasonable results. The initial model should contain as much information about the area under research as possible. This means incorporating information as for example well logs, geological models and, magnetic- and gravimetric measurements if possible.

Well logs contribute to the most accurate source of vertical resolution of geophysical information. The disadvantage of well logs is the very limited horizontal resolution. This is why it is important to use some sort of geo-statistical interpolation to try to obtain a higher resolution. Seismic imaging has a lower resolution than the well logs, but has a greater horizontal extent. It can therefore be of great importance to incorporate these two to gain as much information as possible. This is somewhat difficult due to the different nature of the seismic data and the well-logs. The difference in scale and properties of the seismic signal and the well logs are significant. The well logs are typically very high frequent (in the order of KHz) while the seismic signal is much lower (in the order of 5-100 Hz).

This means that the very low and the very high frequencies of the model are in the null space, which is unconstrained by the seismic data. It has been suggested that this problem can be solved by using fractals (Sen, 2010). Fractals can be defined as self-repeating patterns over different scales. This means that a fractal will look the same from a far distance as from a close distance. The structure does not need to be exactly similar on all scales, but need to exhibit the same type of structure on all scales.

Sen et al., (2010) argues that a reflectivity sequence also follows this fractal behavior, unlike the common assumption of whiteness. It is also assumed that the power spectrum, variogram and covariance of the well logs follow a power law behavior with a scaling exponent in terms of the Hurst coefficient. The Hurst coefficient will be defined in the next section. The idea is that the random reflectivity series we create in our synthetic model are influenced by fractal

Gaussian noise and not completely random noise. The fractal Gaussian noise can be derived from the well logs.

Fractional Gaussian noise can be described as the increment process of a fractional Brownian motion. Brownian motion can be described as a random process but with the properties of: stationary, independent increments and that it have a finite standard deviation. When this is derived from the well logs the probability density function of the fractional Gaussian noise represents variation in subsurface properties. Other advantages of the initial model created in this way are that the parameters are high resolution estimates and that they are in the same frequency range as the well logs**.** The fractional Gaussian noise process generates fractal noise such that the output time series follow the mean and autocovariance of the input series, which naturally are the well logs (Sen, 2010).

If we take a time series which is sampled by a uniform sample spacing in time, Δt, with random normal distributed parameters, and a zero mean. Then the autocorrelation can be defined as (Caccia, 1997):

$$Au(\tau) = \frac{1}{2}\sigma^2[|\tau+1|^{2H} - 2|\tau|^{2H} + |\tau-1|^{2H}] \qquad (3.31)$$

Where σ^2 is the variance of the process, τ is the time separation of the random variables and H is the Hurst coefficient. $Au(\tau)$ represent therefore the autocorrelation between two of the random samples with a time separation of τ.

The Hurst coefficient may be referred to as the “index of long-range dependence”. It is a number between 0 and 1. When 0< H <0.5 the time series is said to be negatively correlated, meaning that a large value will probably be followed by a low value. When 0.5<H<1 the time series is said to be positively correlated meaning that a high value will probably be follow by another high value. Since the Hurst coefficient is an expression for long term dependencies it is plausible that this tendency may go on for quite some time. As we can see from equation (2.10) if the Hurst coefficient is 0.5 it is said to be long term uncorrelated, in other words Gaussian white noise.

The Wiener-Khinchin theorem says that if we take the Fourier transform of the autocorrelation we get the power spectral density of a wide-sense stationary random process (Sen, 2010).

The way of generating a fractional Gaussian time series $\{Y_t\}$ of length N when we have the H from the well logs, can be described as follows (Sen, 2010):

The power spectrum can be calculated by a discrete Fourier transform of equation (2.10):

$$S_k = \sum_{\tau}^{M/2} Au(\tau)e^{-i2\pi k(\frac{\tau}{M})} + \sum_{\tau=\frac{M}{2}+1}^{M-1} Au(M-\tau)e^{-i2\pi k(\frac{\tau}{M})} \quad , \qquad (3.32)$$

If we let N be a power of 2, and we let $M = 2N$. Then for $k = 0,1,2 \ldots . M/2$.

It is important to check that the spectrum is positive for all values of k, or the series are invalid. Caccia (1997) mentions that negativity has never been observed, but that it does not exist a proof that it could not occur.

If we then let W_k be a set of independent and identically distributed random variables with zero mean and unit variance the randomized spectral amplitudes can be written as:

$$V_0 = \sqrt{S_0}W_0 \qquad (3.33)$$

$$V_k = \sqrt{\frac{1}{2}S_k}\,(W_{2k-1} + iW_{2k}) \quad for\, 1 \leq k < \frac{M}{2} \qquad (3.34)$$

$$V_{\frac{m}{2}} = \sqrt{S_{\frac{M}{2}}}\, W_{M-1} \qquad (3.35)$$

$$V_k = V_{m-k}^{*} \qquad (3.36)$$

Where * denotes complex conjugate.

Finally, we used the first N elements of the discrete Fourier transform on V_k to compute the time series:

$$Y_c = \frac{1}{\sqrt{M}} \sum_{k=0}^{M-1} V_k e^{-i2\pi k(\frac{c}{M})} \quad Where\ c = 0,1, \dots \dots N-1 \tag{3.37}$$

In this way we are able to produce a reflection time series with noise which does not have a completely random nature, but is longtime correlated with data from the well logs.

It can also be showed that statistics of interest in fractal Gaussian noise are related. This includes mean, variance, spectra and fractal dimension. The relationship can be written:

$$A(f) \propto \frac{1}{f^{\beta}} \tag{3.38}$$

Where $A(f)$ is the spectrum, f is the frequency and β is defined as:

$$\beta = 2H - 1 \tag{3.39}$$

For the case of white noise, equation (3.38) shows that the spectra are constant. This because when H = 0.5, the fractal Gaussian noise turns into Gaussian noise by equation (3.31) and β = 0, thereby giving a constant amplitude spectra in equation (3.38) (Caccia, 1997).

3.6.1 Computing the Hurst coefficient

If the number of samples is sufficient, the Hurst coefficient can be calculated using rescaled range analysis. Many dataset in nature follow a power law on the form of (Sen, 2010) :

$$\frac{R^*}{S^*} = (\frac{N}{2})^H \qquad (3.40)$$

Where N is the number of samples, S^* is the standard deviation and R^* is the running sum range defined as:

$$R_n^* = (Y_n)_{Max} - (Y_n)_{Min} \ , \qquad (3.41)$$

where Y_n is the function being investigated.

Equation (2.19) can easily be used to find H by plotting it in a log-log scale where H would be the gradient (Caccia, 1997). So by using the rescaled range method we can estimate a Hurst coefficient from our impedance data and thereby be able to get an expression for the long term “memory” of the time series. This is mainly done by inverting the seismic data into impedance information, both acoustic and shear impedance.

4 Simulated Annealing

Simulated annealing is a method used in solving optimization problems. It uses the analogy from the process of slowly cooling a melt until a crystal is formed. This happens when the substance reaches its lowest energy state. It is important that the cooling is conducted very slowly or a metastable state may form. The procedure can be thought of as finding a global minimum where many local minima are present. If the cooling happens too quickly it represents finding a local minimum (Tarantola, 1987).

Simulated Annealing is a Monte Carlo method which uses a probability density function in order to make reasonable guesses in the neighborhood of an initial model. If a random realization represents a better fit than the initial guess, the properties of the model are updated, and new random realizations are made.

To be able to go “uphill” and escape from local minima it uses the Metropolis criteria in equation (3.29). Simulated Annealing can be used in all optimization problems where the goal is to find a global minimum. Many geophysical problems, including model based inversion problems endeavor to find the global minimum of the error between the modeled synthetics and the real seismic data.

4.1 Mathematical Background

The Mathematical background of simulated annealing is an adjustment of the Metropolis algorithm. While the Metropolis algorithm remains a constant temperature, thereby simulating the average behavior of a physical system in thermal equilibrium, Simulated Annealing lowers the temperature gradually. As number of iterations increase, the temperature decreases gradually with the result that the probability of making an uphill move also decreases. The constant temperature of the Metropolis algorithm makes the same change in energy has the same likelihood of accepting an uphill move regardless of how many iterations that has been made, simulated annealing gradually decreases the probability of accepting a uphill move as the algorithm goes towards the end. This makes the algorithm only accept perturbations which lower the energy, as the temperature goes towards zero.

The algorithm works in the following way (Rothman, 1985):

- Choose an initial model
- Choose an initial temperature

1) Make a Perturbation of the initial model with a random variable X
2) Calculate the error of the new parameters, if the error is smaller than the previous accept the new model as answer
3) If the error is larger than the previous, accept according to the Metropolis criteria
4) Lower the temperature, repeat from step 1

The random variable we make the perturbation to our parameters with can be described in the following form:

$$X = rv \tag{4.1}$$

Where r is a uniform random number between -1 and 1 and v is a scaling vector, or increment size. By doing this we are upholding the Markov chain criteria since the next variable is only dependent on the previous. The choice of scaling vector is highly dependent on the parameters we want to compute and should be chosen wisely. It is common that the scaling vector is reduced dramatically toward the end of the routine in order to find the minimum in the neighborhood of the current best parameters.

The method has the advantage of all other Monte Carlo methods that the relationship between the synthetic data and the real data do not need to be linear. Because of these relaxed assumption, the algorithm can more easily deal with very complicated functions.

4.2 Cooling Schedule

With cooling schedule we mean the rate at which the temperature is lowered. If the temperature is lowered too quickly we may miss the global minimum and get "trapped" in a local minimum instead. (Rothman, 1985) The most effective cooling schedule is starting up with a high temperature. This enables our initial guess to be very widespread and can therefore "detect" possible minima in the entire search space. If our probability distribution is the Gaussian or the Gibbs distribution, the initial very high temperature is more or less the equivalent of a uniform purely random distribution. When the temperature decrease sufficiently this is ensuring that we narrow our search around the current optimum. As the temperature reaches zero the distribution is starting to look like the gradient ascent assuring that we find the absolute minimum searching in the neighboring solutions (Rothman, 1985). The rate of which we cool down the temperature is very important in order to be sure that we are converging to a probability of 1 of finding the global minimum. Geman and Geman (1984) proved that if the temperature is lowered by the formula:

$$T(k) \geq \frac{c}{\log(1+k)} \qquad (4.2)$$

Where k is number of realizations and c is a constant independent of k.

As $k -> \infty$ the probability of finding a global minimum will converge to 1. This sounds good in theory but the logarithmic decrease of the temperature may also be to slow to be feasible in an inversion problem. For example, after 1 million iterations, the temperature is only lowered by a factor of 6. If the natural logarithm is chosen it is lowered by a factor of approximately 14. If a linear approximation is assumed it is lowered by a million. Rothman (1985) suggested that the temperature should initially be lowered on the form

$$T_k = T_0 0.99^k \ , \tag{4.3}$$

where k is the current number of iterations. This should be used in the start of the routine examining the search space before the logarithmic temperature decrease in equation (4.2) takes over to assure slow enough decrease to enable annealing.

Which cooling schedule that should be used is very case specific. In the example of seismic inversion, using the forward model in section 2.2 the error function is the difference between the observed reflection coefficient and the calculated reflection coefficient. These have the range between -1 and 1 and their difference may be very low. The greatest difference it can have is 2, and by choosing the initial temperature in this area, we should be sure that the temperature is sufficiently high for the possible exploration of the complete model space. However, we also need to make sure that the temperature is lowered sufficiently. If the number of iterations is chosen to be one million, and the cooling schedule is natural logarithmic, the end temperature would be approximately 0.14 with 2 as initial temperature. This temperature may be too large and it should be investigated if this is sufficient, or if the temperature should be lowered at a faster rate in the beginning. An increase in the number of iterations is also a possibility though it would increase the computational time. These are questions that need to be addressed.

4.3 Pitfalls

Simulated Annealing is a very effective method of finding a global minimum if used correctly. Where other inversion schemes often find local minima certain variants of Simulated Annealing will find the global minimum. Problems occur however when the exact solution is not the global minimum, i.e. not the solution that gives the least error between the synthetic and the real data. The reason for this is the fact that the forward modeling procedures are approximations, the real data contain noise and also numerical limitations of the routine. Other problems are non-uniqueness, several solutions which gives the same error as the correct solution. It may be very difficult to distinguish these. This applies for all global inversion routines.

As seen in chapters 3.3 and 3.6 we can try to reduce the number of solution by setting constraints on our parameters and incorporate all prior information in a sensible way. This can be done by incorporating maximum and/or minimum acoustic impedance limits, velocity limits, density limits or other constraints regarding the maximum allowed change from our initial guess. This also creates a problem that our solution is “colored” by our initial guess. If the correct solution is not in our range of constraints, we will never find it no matter which inversion routine we use.

5 Synthetic modeling

Simulated Annealing can be used in various optimization problems. In the following section there will be showed how simulated annealing can be used to invert a seismic dataset. An earth model has been created and the reflection coefficients for different angles have been calculated. These have been calculated following the forward model in section 2.2. A simulated annealing routine has been made with the intension of inverting these reflection coefficients back to rock properties. The Simulated Annealing routine has been programmed in Matlab and can be found in Appendix D. The routine can in principle be used in any optimization scheme.

Since the initial temperature, the initial parameters and the cooling schedule all effects the routine and its ability to detect a global minimum the routine will be demonstrated on a simple forth order equation with two minima in order to clearly see which properties that affect the routines ability to find the global minimum.

5.1 Example of Optimization Problem

First, it will be demonstrated that simulated annealing can be used in any optimization problem. The equation:

$$Y(X) = X^4 + 2X^3 - 98X^2 + 2X + 1 \quad (5.1)$$

Has the model space seen in figure 2.

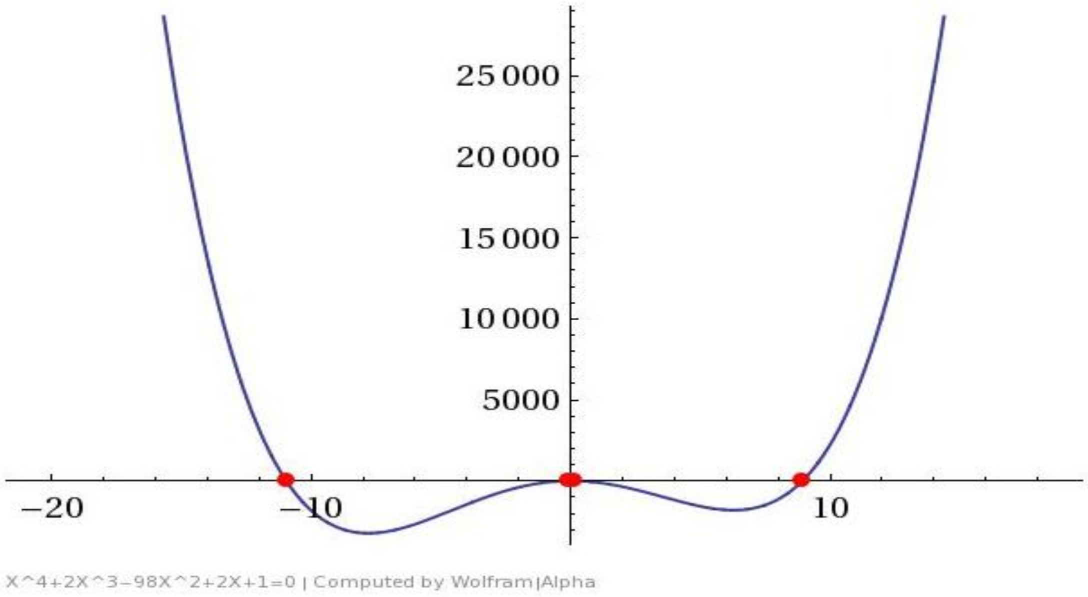

Figure 2: A graph of equation (5.1)

As seen from the figure this equation has two minima. The simulated annealing routine has been tested on equation (5.1) to find the global minimum. The global minimum is located at approximately X = -7.79462. The local minimum is located on the positive x-axis at approximately X = 6.7.

The algorithm will be run by trying several different initial guesses, different temperatures, different increments and different number of iterations, and see how these reflect the algorithms ability to locate a global minimum. The perturbations have been chosen uniform randomly, and the Metropolis criteria have been implemented. The simulation was done in Matlab and the Matlab code can be found in Appendix B. The algorithm was run 1000 times with each parameter setting and it was then observed how many times the routine found the global or the local minimum.

The cooling schedule was chosen to be logarithmic as seen in equation (4.2). After 90% of the iteration the increments were reduced by a factor 10 in order to search in the neighborhood of the current minimum

This procedure assured that the global minimum always was found with an accuracy of ± 0.01. All runs are done with an initial guess of values x = -15, -10, -5, 0, 5, 10 and 15. The values of -15 and + 15 lie in a strongly dipping slope on the left side of the global maximum, and on the right side of the local minimum, respectively. The values of -10 and -5 are located near the global maximum, while the values of 5 and 10 are located near the local maximum.

First, 1000 iterations or random guesses were made for each run. Since the global solution is assumed to be found when the solution lie in an interval of ± 0.01 of the actual solution, the equivalent full model space exploration would be to search through the X range from -20 to 20 with an interval of 0.02 which comprises of 2000 iterations. Half of this should be a good place to start the Simulated Annealing algorithm.

5.1.1 Increment Size

First we will take the increment size of equation (4.1) into consideration. We need to find out how important the choice of increments size is to the routines ability to find the global minimum. We will start with a low increment size and increase it gradually until all the runs finds the global minimum. The algorithm was run with an initial temperature of 100, decreasing logarithmically as a function of iterations, described by function (4.2) with c being the initial temperature.

Table 1: Increment 5, Temperature 100

Number of iterations	Increment	Initial Parameter X	Global Minimum Obtained	Other value obtained	Local Minimum Obtained
1000	5	-15	1000	0	0
1000	5	-10	1000	0	0
1000	5	-5	1000	0	0
1000	5	0	515	0	485
1000	5	5	0	0	1000
1000	5	10	0	0	1000
1000	5	15	0	0	1000

As seen from the table the increment is too small for the algorithm to be able to climb out of the minimum from where it started. When the starting position is X= 0, which is in between the two minima, the number of global minima and local minima obtained are almost equal. The increment size is clearly too small, and the routine success rate is only based on the initial guess of X.

Table 2: Increment 9, Temperature 100

Number of iterations	Increment	Initial Parameter X	Global Minimum Obtained	Other value Obtained	Local Minimum Obtained
1000	9	-15	1000	0	0
1000	9	-10	1000	0	0
1000	9	-5	1000	0	0
1000	9	0	515	0	485
1000	9	5	0	0	1000
1000	9	10	199	0	801
1000	9	15	63	0	937

When the increment size where chosen to 9 we can see that the global minimum where found every single time if our initial guess was in the neighborhood of the answer. This is the same as in table 1, and is to be expected due to the initial values proximity to the global minimum. However if the initial guess was near the local minimum the global solution was found on a small number of runs. It is an increase from when the increment size was 5, but it is very low. When initial guess was x= 10, it was found on approximately 20% of the times, and 6.3% of the times when the initial guess was x = 15. However when x= 5 the global minimum was not found a single time. The reason for this may be that x=5 is the starting value which has a very low value in the local minimum. If the increment size is not large enough, it is highly unlikely that the routine will find a lower value that is not in its neighborhood.

Table 2: Temperature 100

Number of iterations	Increment	Initial Parameter X	Global Minimum Obtained	Other value obtained	Local Minimum Obtained
1000	10	-15	1000	0	0
1000	10	-10	1000	0	0
1000	10	-5	1000	0	0
1000	10	0	626	0	374
1000	10	5	274	0	726
1000	10	10	368	0	632
1000	10	15	142	0	858

By increasing the temperature the global minimum is obtained more often even though the number of iterations and temperature are kept constant. We can see that even this minor adjustment in increment size increases the number of global minima found when the initial model has been chosen in a local minimum. They are indeed a minority but they are found between 10-30% depending on the initial value.

If we were to increase the increment to 12 the global minimum will be found on every single test independent of starting position. The same is the case for increments over 12. What this test basically shows is that for an initial temperature chosen as low as this the Simulated Annealing routine is more or less just a Monte Carlo method with the range of guesses outside the area where the global minimum is located. When the temperature is as low as in this example the Metropolis criteria is probably very seldom fulfilled, and the routine is unable to climb out of the local minimum.

5.1.2 Temperature

It is important for the algorithm to work that the temperature is relatively high in the beginning. If it does not start high enough we may not be able to fulfill the Metropolis criteria in a number of cases. If the Metropolis criteria is not fulfilled the algorithm does not have the ability to climb uphill and thereby escaping local minima. In that case the algorithm acts like a pure random Monte Carlo method where the model parameter is never updated unless a better minimum is found. This was the case in the last section. We need to find out a good starting point for the temperature.

The energy, error or cost function is the function we would like to minimize. In our example it is the function itself because it is this minimum we are interested in. The change in cost function can be described in this example as the difference between the current minimum and the so far best obtained minimum. The global minimum of equation (5.1) has a value of approximately – 3224.52 at x = -7.794. As we can see from figure 2 the values quickly arises to several tens of thousands when the X is increased to around and over 20. We need therefore to have a temperature which is high enough to be able to fulfill the Metropolis criteria when the energy has a difference of at least a couple of hundreds.

Let us say that we are located around the global minimum with an X value of 6. Here the function has a value of -1787. We can then calculate the probability that the algorithm will accept an uphill move to a lower energy level. At X = 3, the minimum is -740.

In the table below you can see the acceptance probability at the different initial temperatures:

Table 3: Temperature versus Acceptance Probability

Temperature	Change in energy divided by temperature	Probability of accepting uphill move
100	-10.47	0.000028375
200	-5.235	0.005326824
500	-2.094	0.123193376
1000	-1.047	0.350989139
1500	-0.698	0.497579438
2200	-0,475	0.621319693
2300	-0.455	0.634310060

From table 3 we can see that the acceptance probability increases dramatically with temperature. This has the effect that the Metropolis criteria are fulfilled more often. With the lower temperatures practically none are accepted, and thereby no uphill moves are made. We need to find out how this affects the algorithm in obtaining the global minimum.

We will now use the same increment and number of iterations as in table 1 but change the starting temperature to see if the global minimum is obtained more often with higher temperatures. The calculations can be found in Appendix A.

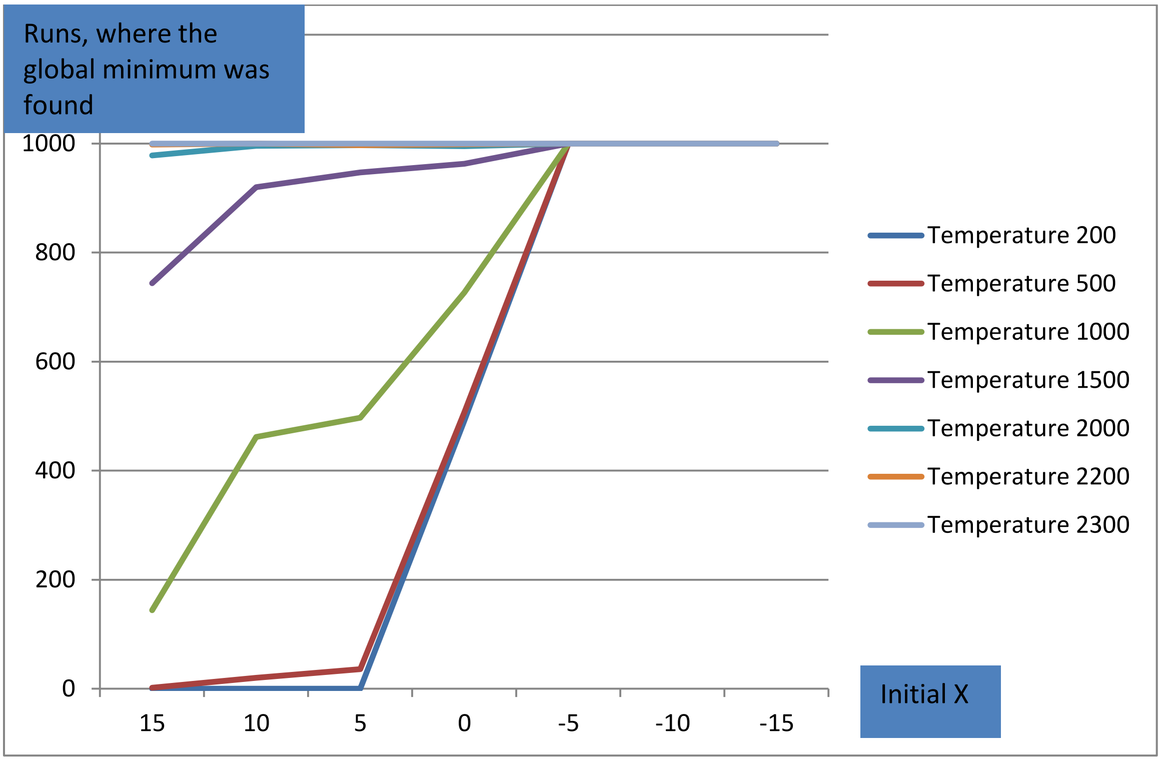

Figure 3: Global minimum found as a function of initial starting value. The different functions represent different starting temperatures.

As we can see the algorithm finds the global minimum when the initial parameter is in its neighborhood regardless of temperature. This is what is to be expected for low temperature since it is strictly speaking just a Monte Carlo method with an increment too low to reach the model space where the local minimum is located. Since the temperature is low, the Metropolis criteria will very rarely be fulfilled.

As the initial temperature increase the number of global minima increases gradually in the area where the initial guess is close to the local minimum. This shows that the Metropolis criteria are fulfilled more often and the algorithm is able to search in the complete model space. This is the same as table 3 suggest. When the temperature is increased to approximately 2200 the global minimum is found regardless of initial parameter. If we go back to our example in table 3, in this particular case the temperature of 2200 would have a starting acceptance probability of 0.62, which means that the trade would be accepted with a 62% probability. This may give us a clue in what order the minimum starting acceptance probability, hence the temperature should be.

Figure 3 shows us that by increasing the initial temperature we find the global minimum in all cases even though the increment is low. This is very easily explained by the fact that the high temperature allows many starting moves to be accepted as the new parameter thereby letting the algorithm search in the entire search space. As number of iterations increase fewer and fewer bad transitions are made, and the algorithm chooses only to move downhill.

This means that the increment size is not a very important factor as long as the temperature is chosen sufficiently high.

Geman and Geman (1984) showed that if we use a logarithmic method in equation (4.2) the probability goes toward 1 of finding the global minimum as number of iterations goes towards infinity. The problem with this method is that the number of iterations that is required before the temperature is lowered sufficiently for the simulated annealing to only search in its neighborhood for the optimum solution may in some cases be enormous. So unless we have a sufficiently number of iterations, we need to implement an algorithm that searches in the neighborhood of the current solution as the run go towards the end. However, the problem is that if there are many parameters involved, this may take a lot of time.

5.1.3 Number of iterations

Then the algorithm was run with the best increment and temperature found in sections 5.1.1 and 5.1.2 and the number of iterations was slowly lowered in able to find how few iterations it was possible to choose and still get the global minimum as the answer. The increment was set to 12 and the initial temperature to 2300. After 90% of the iterations, the increment size was lowered by a factor of 10 for the algorithm to search in the neighborhood of the current minimum. The algorithm found the global minimum on all 1000 runs, regardless of initial guess, down to approximately 150 iterations. Then 1 out of the 1000 runs gave the local minimum as the answer, if the initial parameter was chosen inside the local minimum. This clearly shows the ability of the algorithm to find a global minimum on relatively few iterations.

With this example we have found that if the starting temperature is high enough, this algorithm is able to find a global minimum on a relatively small number of trials. The increment size is not important when the temperature is high, because a great number of the initial perturbations are accepted as new models, thereby letting the algorithm search through the entire model space.

5.2 Seismic Inversion

In this section the inversion routine will be conducted on a more complicated problem. A seismic synthetic set has been made by creating reflection coefficient using the equations showed in section 2.2 regarding the forward problem. These have then been inverted by using Simulated Annealing algorithm quite similar to the algorithm used in section 5.1. In section 5.1 there is however only one unknown parameter and it is therefore easy to model the search space. In the inversion of seismic data we have, however, three unknown parameters and the search space is more difficult to examine.

5.2.1 The model

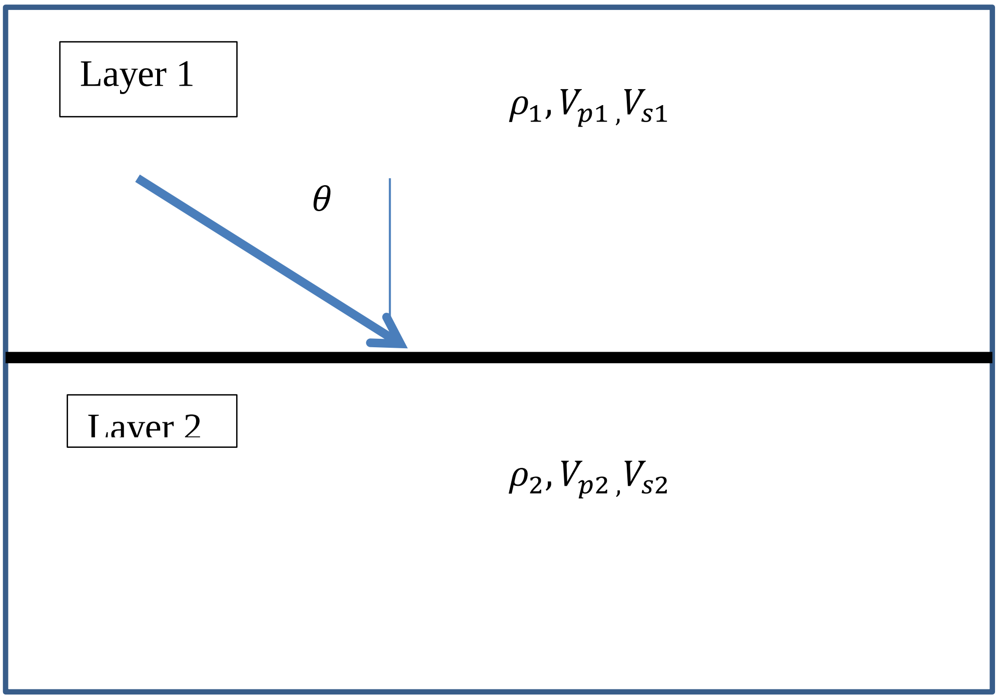

Figure 4: Basic reflection model showing a wave hitting an interface between two solid media

The synthetic model can be illustrated in figure 4:

As shown on the figure this is the most basic layer model we are trying to invert. The ρ represents the density while V_p is the pressure velocity and V_s is the shear wave velocity. The reflection coefficients has been calculated for six different angles representing an angle gather corrected for all other amplitude effects. As showed in the section of forward modeling, we are inverting for three parameters using the linear approximation of the Zoeppritz equation showed in equation (2.6). These are the average P impedance across the interface, the average S impedance across the interface, and the relative change in density.

The properties of the layer were set to the following:

$$\rho_1 = 2100 \frac{kg}{m^3}, V_{p1} = 2800 \frac{m}{s} \; V_{s1} \; = \; 1500 \frac{m}{s}$$

$$\rho_2 = 2500 \frac{kg}{m^3}, V_{p2} = 3200 \frac{m}{s} \; V_{s2} \; = \; 1800 \frac{m}{s}$$

This makes the layers average pressure, shear impedance and relative change in density:

$$Z_p = 6\,940\,000 \frac{kg}{m^3}\frac{m}{s}$$

$$Z_s = 3\,825\,000 \frac{kg}{m^3}\frac{m}{s}$$

$$R_d = 0.17391$$

The reflection coefficients were calculated for the pre critical angles of 2, 10, 15, 17, 20 and 30. The calculated reflection coefficients were then:

Table 4: Reflection coefficients versus angle

Angle	2	10	15	17	20	30
Reflection coefficient:	0.15331	0.14585	0.13653	0.13194	0.12425	0.09419

5.2.2 The Inversion Routine

The zero offset reflection coefficients for pressure waves, shear waves and densities have been taken from the routine that created the model. These have been calculated as the logarithmic approximations in equations (2.4) and (2.5). These reflection coefficients represent reflection coefficients from the seismic dataset, which are completely filtered from all other sources of noise. The only issues regarding these coefficients are the approximations in the equations and the numerical accuracy of the script. The zero offset reflection coefficients were implemented with an accuracy of 15 digits. The routine was run a hundred times, each with 100 000 iterations. The initial guesses were set to:

$$Z_p = 7\,000\,000 \frac{kg}{m^3}\frac{m}{s}$$

$$Z_s = 4\,000\,000 \frac{kg}{m^3}\frac{m}{s}$$

$$R_d = 0.2$$

The only constraints made on the algorithm was that the range of R_d was set to be between 0 and 0.7. This is a very loose constraint and should probably have a smaller range, since the approximations of the Zoeppritz equations are not valid for the outer part of this range. However, since the goal of this inversion is to illustrate a global optimization algorithm the constraints were soft rather than hard. The reason this constraint was implemented was because without any constraint the inversion produced only unphysical solutions. These solutions had negative relative change in density and impedance below 1 $\frac{kg}{m^3}\frac{m}{s}$.

The term that is to be minimized by the algorithm:

$$error = \sum_i^N |r_d\,(\theta_i) - r_m(\theta_i)| \qquad (5.2)$$

Where r_d is the reflection coefficient from the data, r_m is the reflection coefficients generated by the model, θ_i is the current angle of incident and N is the number of angles.

As shown in the previous example these initial models should be irrelevant as long as the temperature is set to be large enough. The initial temperature was set to 0.01, which should be sufficient. This may seem like a low temperature, but the cost function of this problem is a lot smaller than the previous example. Reflection coefficients are in the order of 0.01-0.3. The absolute error between the modeled data and the real data may be as low of magnitude as $10^{-4}$and 10^{-7} and then the starting temperature of 0.01 is relatively large. If we were to have the energy any higher we would risk that the Metropolis criteria was fulfilled on every single iteration, since we have limited number of iterations and the temperature would not have been lowered enough. This would make the routine into just an ordinary random walk Monte Carlo method. The error function was calculated as an absolute value instead of least means square. This was because of numerical limitation of the software. The seismic data which the error estimate is calculated from was implemented with 5 digits of numerical accuracy. The inversion script can be found in Appendix D.

The number of iterations were set to 100 000 to ensure that the model space is searched completely. After 90% of the iterations are run, the algorithms increment size is reduced by a factor of 100 to let the algorithm search closer to its neighborhood for a more optimal solution.

5.2.3 Results

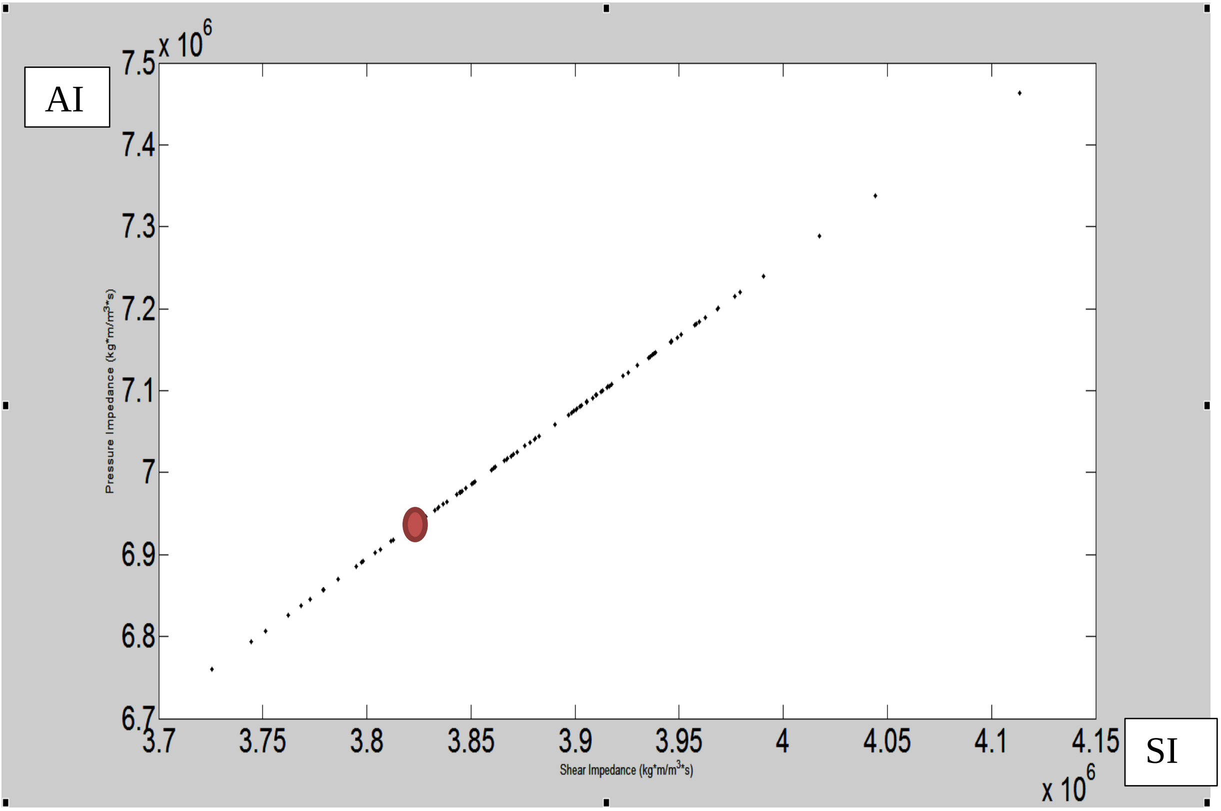

Figure 5: Acoustic impedance (AI) versus shear impedance (SI)

Figure 5 shows the relationship between the hundred different solutions pressure and shear impedances. As we can see they exhibit a linear relationship. This we would assume due to the nature of the forward model. A factorial change in the ratio between the shear and pressure impedance will not change the reflection coefficients. The linear relationship also contains the actual solution as we can see it is marked by the red dot. The problem is however underdetermined and without more information we are unable to obtain the correct solution.

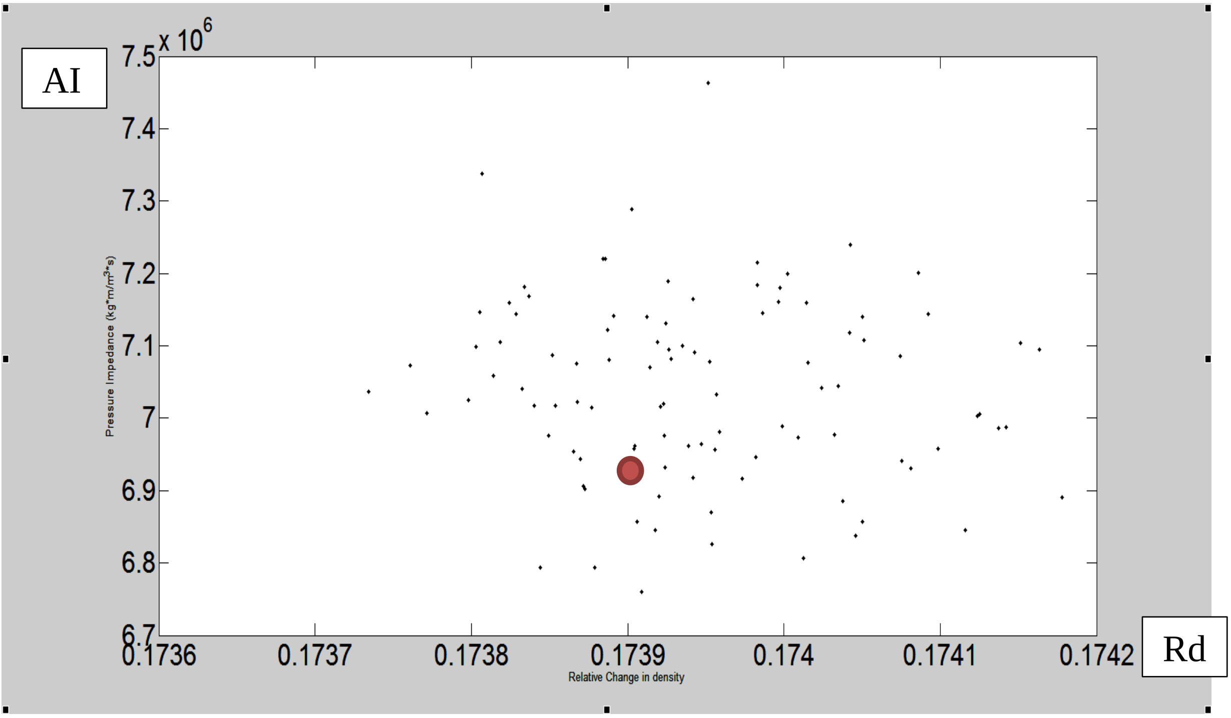

Figure 6: Acoustic impedance (AI) versus relative density (Rd)

As we can see from figure 6 above it does not seems like there are any relationship between the impedance and the relative density. The scatter of solutions looks rather random. The actual solution is marked as the red dot. Here the relative density is approximately 0.17391. A great number of solutions are centered around this area. There are also a number of solutions which has the same relative density but vary greatly in acoustic impedance. To find this solution may therefore be a considerable problem. The advantage, however, is that all the solutions are located inside an area of 0.006 change in relative density. This is a very small change, and all the solutions can be thought of as acceptable with respect to the relative density. If our solution has a relative density of 0.1736 or 0.1742 would not be of particularly much concern.

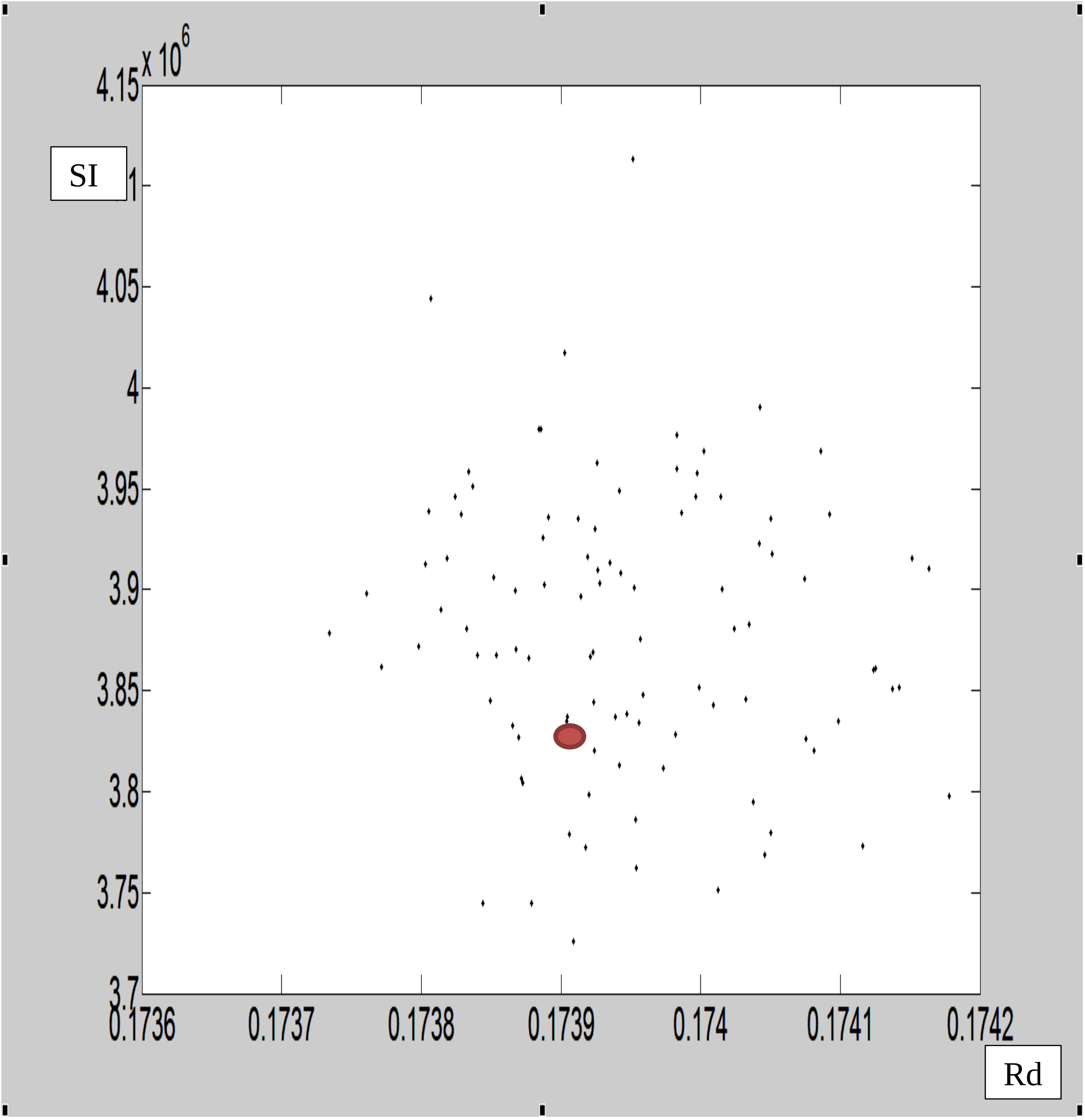

Figure 7: Shear impedance (SI) versus relative density (Rd)

Figure 7 shows that the plot of shear impedance versus relative density is the same exact plot as figure 5 but just scaled differently due to the linear relationship between the shear and pressure impedance. This is what we would expect.

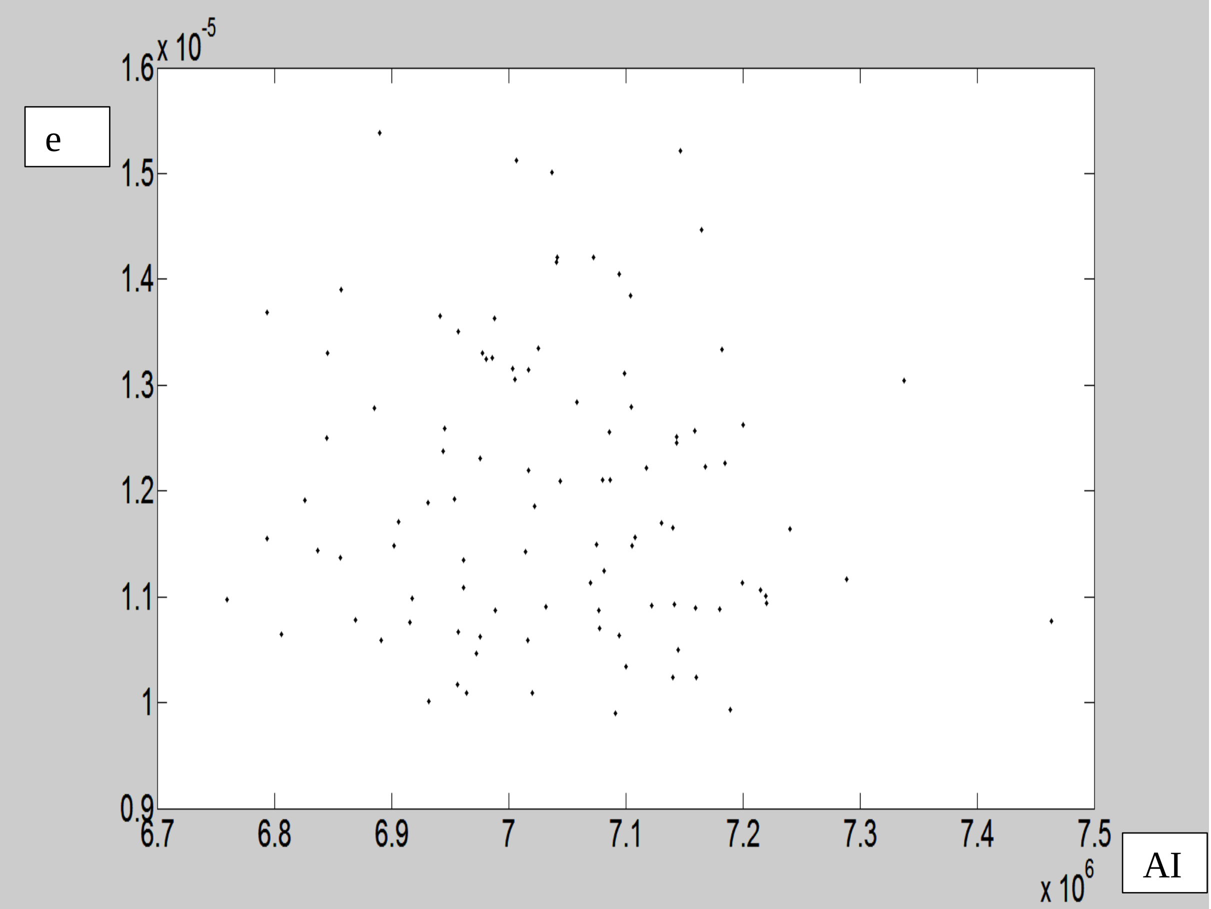

Figure 8: Absolute error (e) versus acoustic impedance (AI)

Figure 8 shows the distribution of acoustic impedance against the absolute error. The two lowest result had an absolute error of $9.9322 \times 10^{-5}$and 9.9322×10^{-5}. These give the pressure impedance of 7.1896×10^{6} and 7.0915×10^{6} $\frac{kgm}{m^3 s}$ respectively. Both of these are quite larger than the correct answer, and because of the linear relationship the shear impedance will also be too large. All of the solutions have an acceptable error if we regard the fact that the numerical accuracy of the input reflection coefficient is five digits and all the solutions are more or less in this area. This leads us back to the fact that all of these answers may represent the correct model, when we only use the information provided by the reflection coefficients. This means that we may have infinite solutions which have a linear relationship between the P and S-impedance. The ratio of the pressure and shear impedance are more or less constant for all solutions. We can see this by equation (2.6) that changing Zp and Zs with the same factor will produce the same result.

5.2.4 Introduction of noise

As seen from the last section we get an infinite number of possible solutions just because of the error caused by numerical accuracy, the introduction of the logarithmic approximation of the reflectivity model and the linear solution space. As we know seismic data contains various sources of noise. We will now try to demonstrate the effects of introduction of random noise to the angle dependent reflections coefficients and see how this will influence the inversion results. There has been made a random perturbation between -0.01 and 0.01 to each reflection coefficient.

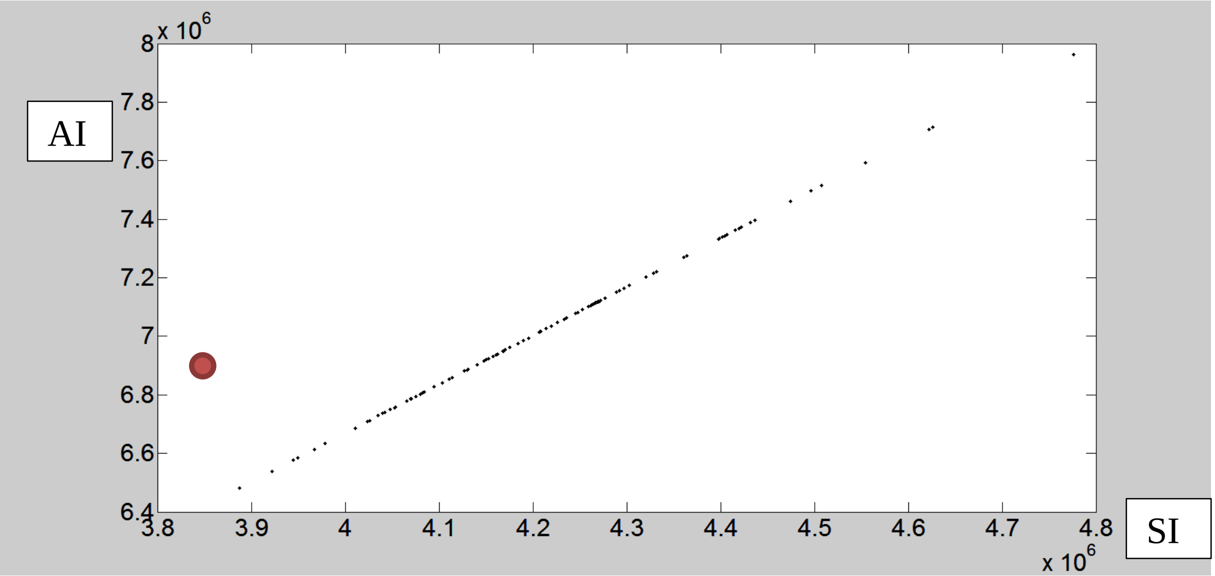

Figure 9: Acoustic impedance (AI) versus shear impedance (SI)

From figure 9 we can see that the solution space still has a linear relationship between the pressure and shear impedance. This is due to the nature of the forward model. The linear relationship are however different from the relationship without noise. We can however see that none of the solutions found are near the correct solution. The correct solution marked by the red dot has considerable larger acoustic impedance than what is suggested from that specific shear impedance.

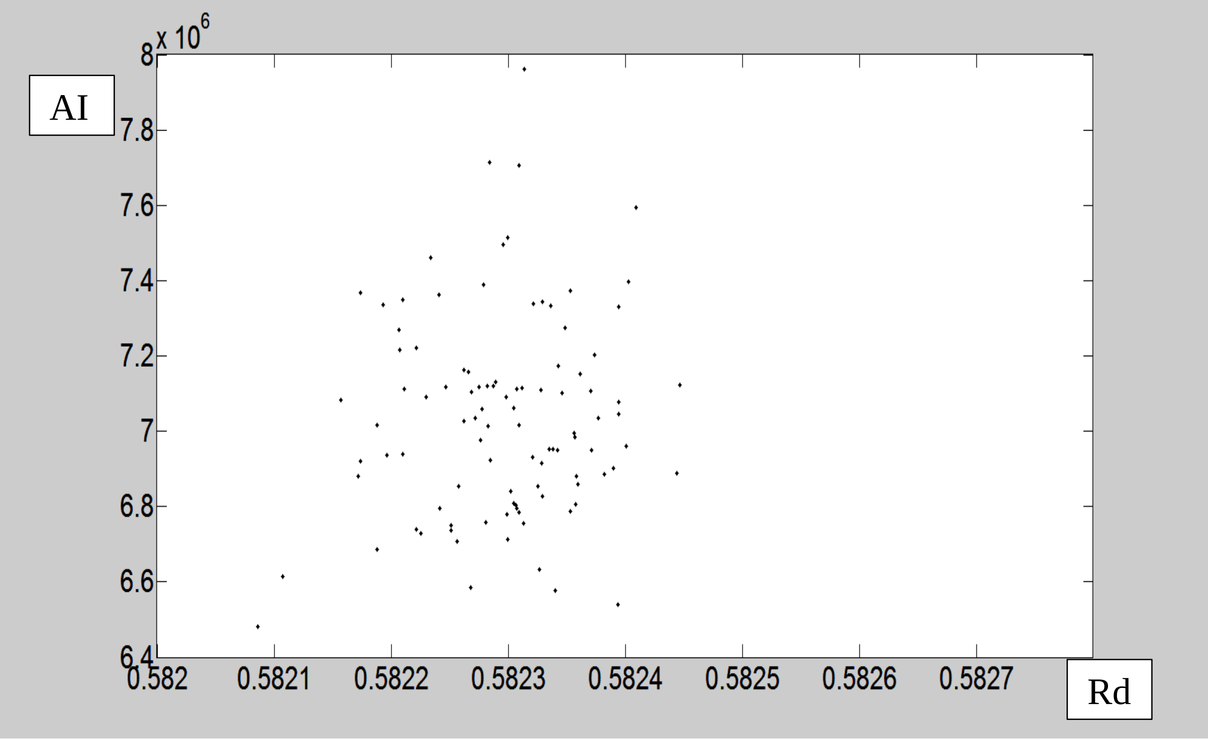

Figure 10: Acoustic impedance (AI) versus relative density (Rd)

Figure 10 shows the relationship between the relative density and the acoustic impedance. Because of the linearity the plot of the shear impedance and the relative density would have the same solution space but been downscaled. As we can see all the solutions are close together in the relative density domain from values of approximately 0.58215 to 0.58245. The problem is that the acoustic impedance in this interval varies drastically from around 6.8 to 8 million $\frac{kg\ m}{m^3 s}$. It is therefore difficult to point out a solution. We can also note that the relative impedance is very large. They are definitely too large for what the linear approximation of the Zoeppritz equations are made for, and their physical meaning should also be investigated.

When we introduce noise we see some of the problems regarding global optimization techniques. First, we have 100 different solutions which all exhibits more or less the same error. The inverted parameters vary hugely in range so we need an interpretation of all the solutions in order to establish the most likely. Secondly, when we introduced noise, all the solutions moved from the correct relative density of around 0.17, to around 0.58. With respect to the change in the relative density, all of our solutions are thereby completely wrong. A relative density of 0.58 would suggest that layer number 2 has almost twice the density as layer number 1. This is in most cases unrealistic. However, all of our solutions which apparently have the lowest error between the seismic data and the synthetic data suggest that this is the best solution. Since all the solutions we have found are unrealistic and wrong, we need to constrain the data, in order to see if the correct solution can be found.

5.2.4.1 Constraining the noise

The constraint set to the algorithm was that the relative density has to be between the intervals of 0 and 0.25.

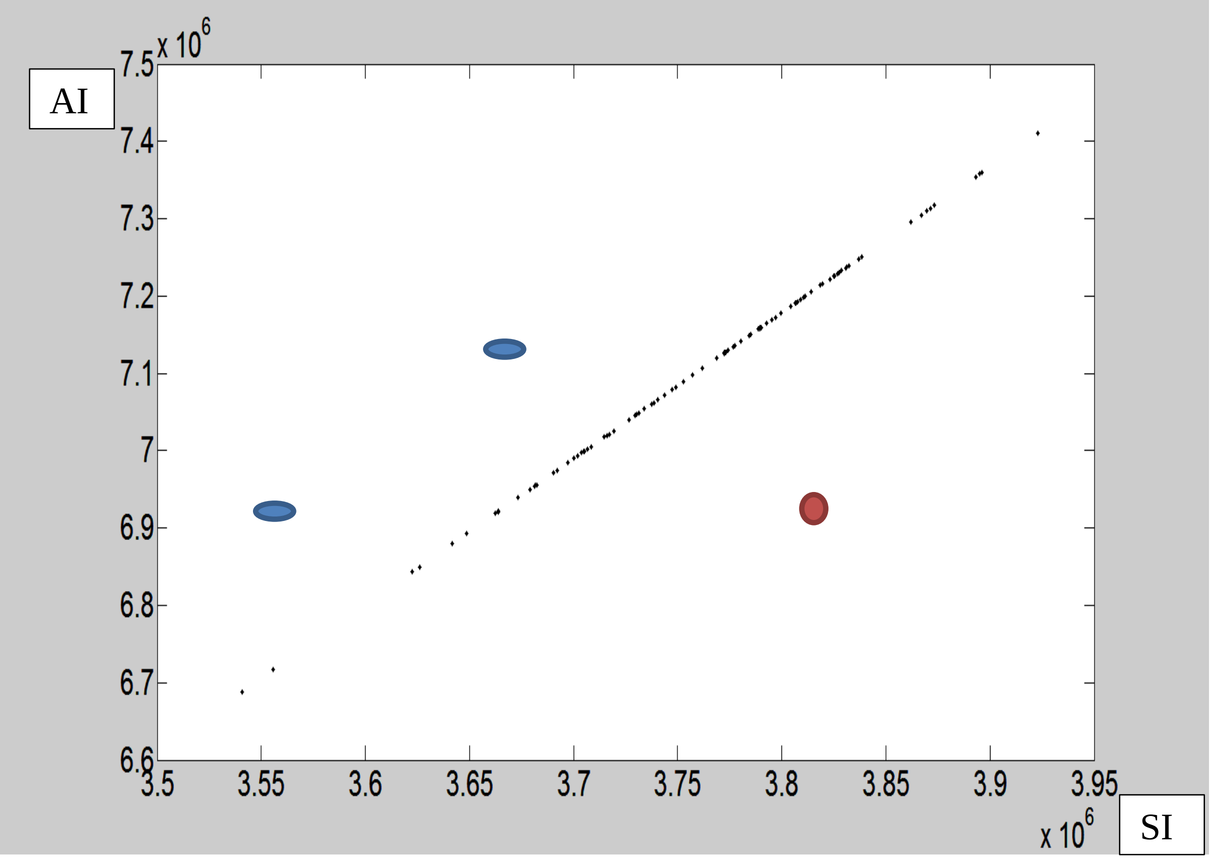

Figure 11: Acoustic impedance (AI) versus shear impedance (SI)

Figure 11 shows the linear relationship between the acoustic and shear impedance for the solution space when the relative density has been constrained. The correct solution has been marked by the red dot, and we can see that the routine has not been able to produce the correct solution. The shear impedance is too low for that specific acoustic impedance.

We can also see two solutions which does not have the same linear relationship with the rest of the solutions. These have been marked by blue dots. It may also seem that these have a linear relationship with each other. This may indicate that these two solutions have found another minimum than the rest of the solutions. In order to find out if this is the global or a local minimum we need to look at the error of these two solutions.

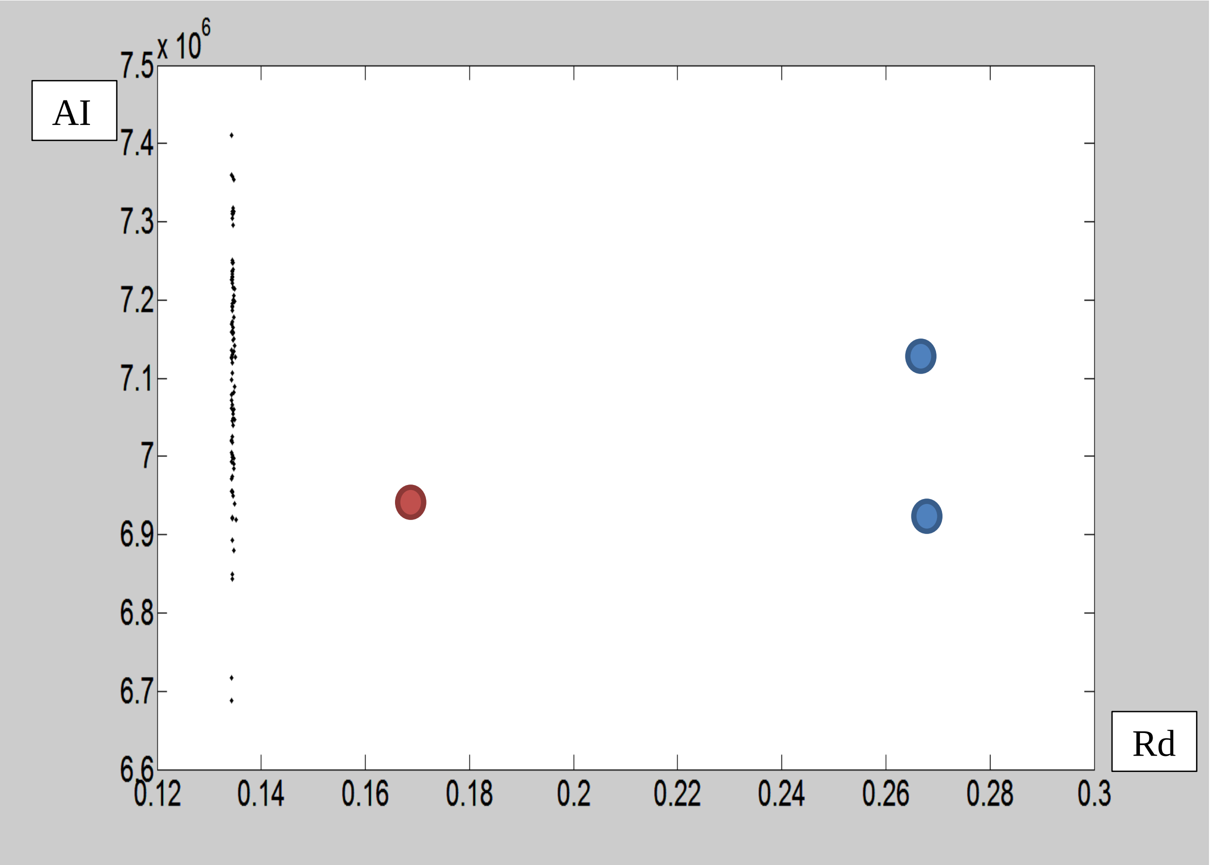

Figure 12: Acoustic impedance (AI) versus Relative density (Rd)

From figure 12 we can see the relationship between the acoustic impedance and the relative density for all the solution. We can see that all the solutions are in more or less the same area regarding the relative density except for two solutions marked in blue. These are the same solutions which did not follow the same linear relationship when plotted as acoustic impedance against shear impedance as the other solutions. We can see that these two solutions have approximately the double relative density then the rest of the solutions.

Regardless, we can see that the correct solution with respect to relative density, which is marked by the red dot, is not found by any of the hundred runs.

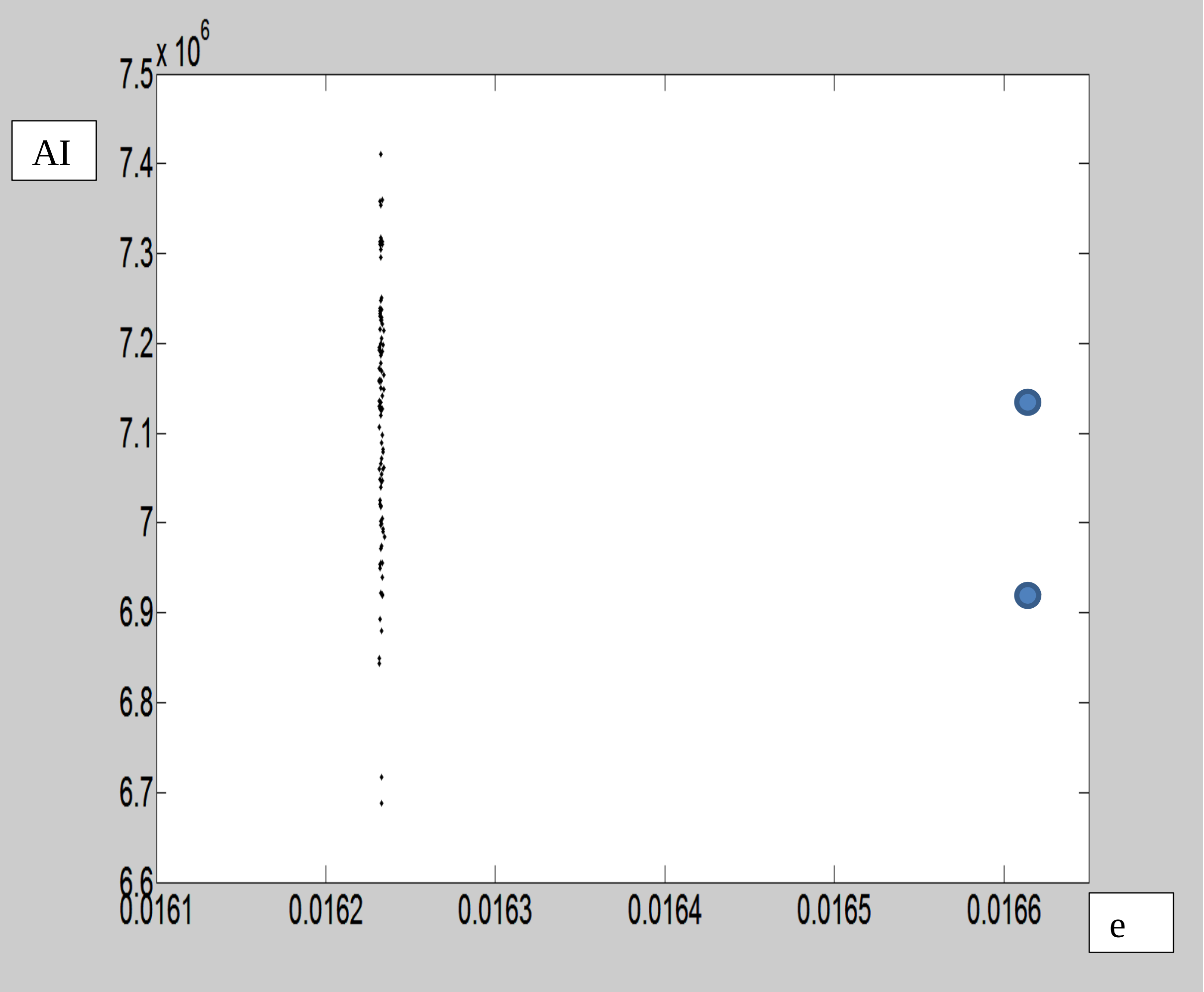

Figure 13: Acoustic impedance (AI) versus Absolute error (e)

We can see from figure 13 that the two samples with the higher estimated relative density, and which does not have a linear relationship with the other solutions in the P-S impedance domain, has a bit larger error than the other solutions. This means that the algorithm has not been able to find the global minimum value on 2 of the 100 runs, and been trapped in a local minimum. To have avoided this it can be discussed if the temperature or the number of iterations should have been increased.

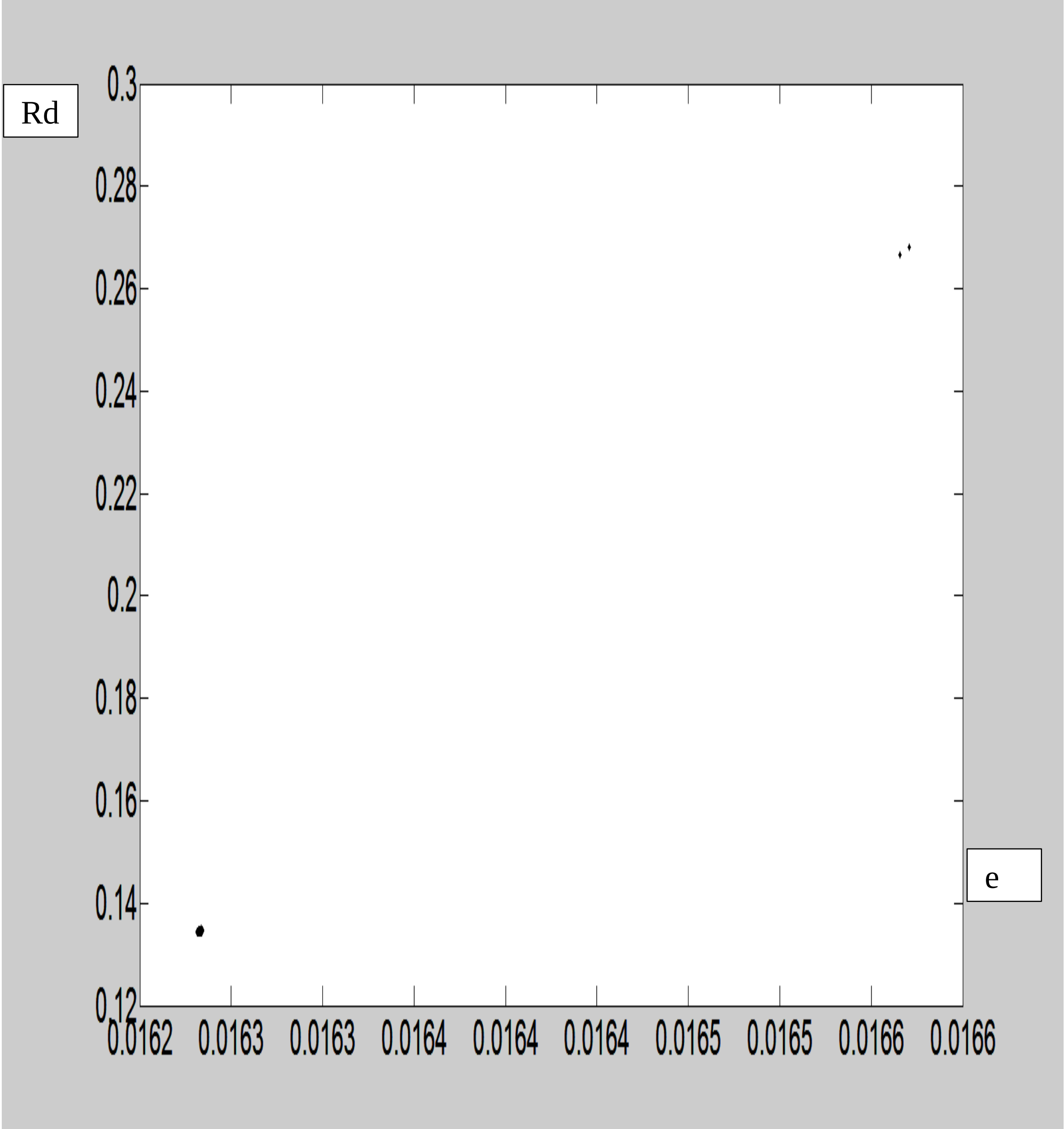

Figure 14: Relative density (Rd) versus Absolute error (e)

Figure 14 tells us the same as figure 13. Two of the runs computed solutions where the relative density was found to be too high, and the absolute error did not match the other 98 solutions.

6 Discussion

In section 5.1 we showed how Simulated Annealing can be used to find a global minimum where local minima are present. As long as the temperature is chosen correctly, allowing the routine to make uphill moves, the optimal solution is found regardless of initial parameters.

When the routine was implemented on the forward model, problems quickly arose. We went from estimating one to three parameters, and the problem became underdetermined because of the linear nature of the forward model. The reflection coefficients computed does not change as long as the ratio between the acoustic and shear impedance are constant. This introduces non-uniqueness immediately, and we are essentially only inverting for two parameters, the relative density and the ratio between the impedances.

When the inversion was conducted on an uncontaminated dataset, the correct solution was found as one out of the hundred runs. All the solutions had a linear relationship in the impedance plot, while the relative density was scattered over a small interval. The inversion algorithm successfully inverted for the relative density and the ratio between the two impedances.

When random noise was implemented we saw how this dramatically affected the routine. At first, a completely unconstrained run was made. The result was that all the impedances where in the order of $10^{-3}$and the relative density was negative. These are clearly unphysical solutions. However, all of them had more or less the exact same low error estimation. The algorithm needed constraints on the relative density in order to make reasonable results. There was not the same need to constrain the impedances, because it was the negative relative densities which gave unrealistically low impedances.

The relative impedance was constrained to a range between 0 and 0.7. The global minimum of this inversion had still too high relative densities with around 0.58. It was clear that the inversion algorithm needed harder constraints.

The relative density was then constrained to an interval between 0 and 0.25. When this constrain was implemented the routine produced two solutions which were local minima, while the other 98 represented global minima. This may suggest that the starting temperature was set too low, or the number of iterations should have been higher. The great majority of the solutions had a relative density of around 0.14. This is a much better estimate than the previous loser constrained inversion. The ratio between the acoustic and shear impedance are found to be significantly higher than it should be, giving a wrong picture of the acoustic and shear impedance relationship.

The inversion of data contaminated by noise confirms that the forward model is very sensitive to changes in the reflection coefficients. By introducing errors the global minimum is very quickly not the correct solution. A minor change in the reflection coefficients, which are between -1 and +1 may change the ratio between acoustic and shear impedances. The change in this ratio may affect the impedance drastically since they are in order of million $\frac{kg}{m^3}\frac{m}{s}$. As we can see the quality of the inversion increase as we add more information through constraints on the parameters. This shows that the constraints on the parameters influence and color the inversion result considerably.

.

7 Conclusion

As seen from the previous chapters we have showed how optimization problems can be solved using the global optimization routine, Simulated Annealing. The theoretical advantage of such a routine is that it should be capable of finding a global minimum, where the model parameters and the data have a non-linear relationship independent of the initial parameter.

In theory, the error between the real seismic data, and the synthetic seismic data should be zero when the correct solution is found. In our case, the difference between the computed reflection coefficients and the coefficients created from the model should therefore be zero. This was not the case in any of the experiments. Even when the data was not affected by noise, the numerical accuracy of the program prohibited the real solution to have zero error. In this case the global minimum was the correct solution, but due to the linearity of the forward model we are only able to invert for two parameters and not three.

The success of Simulated Annealing is highly dependent on the initial choice of temperature, the rate of decrease of the temperature and number of iterations. Even when an initial guess is made completely in the dark, and completely wrong, this routine will be able to produce the global minimum on a great majority of the runs as long as the other parameters are chosen wisely.

Due to the effects noise and mathematical approximations has to the inversion it is not guaranteed that the global minimum will be the correct solution. The forward model is very sensitive to small changes in the reflection coefficients, and this may alter our inversion parameters completely.

To be able to solve these problems in seismic inversion we are in need of more information. As seen in the synthetic example, the inversion results improve as more information is given about the model parameters through the constraints. It is important that this information is implemented in a correct manner with respect to the inversion. This thesis has described how this can be done, and showed the importance of it.

8 References

Aki, K. a. (1980). *Quantitative seismology, Theory and methods.* W.H. Freeman & Co.

Caccia, D. P. (1997). *Analyzing exact fractal time series: evaluating dispersional analysis and rescaled range methods.* Seattle.

Dahl, T. (1990). *Pre Stack Inversion Of Seismic Data.* Trondheim.

Fatti, J. (1994). Detection of gas in sandstone reservoir using AVO analysis. *Geophysics,59*, 1362-1376.

Geman, G. a. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. PAMI-6, NO. 6*, 721-741.

Javanainen. (2009, 12 17). Hentet 06 04, 2012 fra PHYS-5500, Statistical Mechanics: http://www.phys.uconn.edu/~jj/book_5500_v060.pdf

Ma, X.-Q. (2002). Simultaneeous inversion of prestack seismic data for rock properties using simulated annealing. *Geophysics vol 67, NO 6*, 1877-1885.

Rabben, T. E. (2009). *Non-linear Bayesian inversion of seismic reflection amplitudes.* Trondheim.

Rothman, D. H. (1985). Nonlinear inversion, statistical mechanics, and residual statics elimination. *Geophysics, Vol 50, NO 12*, 2784-2796.

Sambridge, M. (2002). *Monte Carlo Methods In Geophysical Inverse Problems.*

Sen, R. S. (2010). Stochastic inversion of prestack seismic data using fractal-based initial models. *GEOPHYSICS VOL 75, NO 3*, R47-R59.

Smith, G. a. (1987). Weighted Stacking for Rock Property Estimation and Detection of Gas. *Geophysical Prospecting, 35* , 993-1014.

Stoffa, M. S. (1995). *Global Optimization methods in geophysical inversion.*

Tarantola, A. (1987). Inverse Problem Theory. Amsterdam: Elsevier.

Ursin, B. &. (1996). The information content of the elastic reflection matrix. *Geophysical Journal International*, *125*, 214-228.

Walsh, B. (2004). *Markov Chain Monte Carlo, Lecture Notes for EEB 581, version 26.* Hentet 06 7, 2012 fra http://web.mit.edu/~wingated/www/introductions/mcmc-gibbs-intro.pdf

Zoeppritz. (1919).

9 APPENDIX

APPENDIX A

This appendix shows the results from the Simulated Annealing routine, where equation 5.2 was inverted.

Table 5: Increment 5, Temperature 100

Number of iterations	Temperature	Initial Parameter X	Global Minimum Obtained	Other value obtained	Local Minimum Obtained
1000	200	-15	1000	0	0
1000	200	-10	1000	0	0
1000	200	-5	1000	0	0
1000	200	0	492	0	508
1000	200	5	0	0	1000
1000	200	10	0	0	1000
1000	200	15	0	0	1000

Table 6: Increment 5, Temperature 500

Number of iterations	Temperature	Initial Parameter X	Global Minimum Obtained	Other value obtained	Local Minimum Obtained
1000	500	-15	1000	0	0
1000	500	-10	1000	0	0
1000	500	-5	1000	0	0
1000	500	0	507	0	493
1000	500	5	36	0	964
1000	500	10	20	0	980
1000	500	15	2	0	998

Table 7: Increment 5, Temperature 1000

Number of iterations	Temperature	Initial Parameter X	Global Minimum Obtained	Other value obtained	Local Minimum Obtained
1000	1000	-15	1000	0	0
1000	1000	-10	1000	0	0
1000	1000	-5	1000	0	0
1000	1000	0	727	0	283
1000	1000	5	497	0	503
1000	1000	10	462	0	538
1000	1000	15	144	0	856

Table 8: Increment 5, Temperature 1500

Number of iterations	Temperature	Initial Parameter X	Global Minimum Obtained	Other value obtained	Local Minimum Obtained
1000	1500	-15	1000	0	0
1000	1500	-10	1000	0	0
1000	1500	-5	1000	0	0
1000	1500	0	963	0	37
1000	1500	5	947	0	53
1000	1500	10	920	0	80
1000	1500	15	744	0	256

Table 9: Increment 5, Temperature 2000

Number of iterations	Temperature	Initial Parameter X	Global Minimum Obtained	Other value obtained	Local Minimum Obtained
1000	2000	-15	1000	0	0
1000	2000	-10	1000	0	0
1000	2000	-5	1000	0	0
1000	2000	0	995	0	5
1000	2000	5	997	0	3
1000	1500	10	996	0	4
1000	1500	15	978	0	22

Table 10: Increment 5, Temperature 2200

Number of iterations	Temperature	Initial Parameter X	Global Minimum Obtained	Other value obtained	Local Minimum Obtained
1000	2200	-15	1000	0	0
1000	2200	-10	1000	0	0
1000	2200	-5	1000	0	0
1000	2200	0	999	0	1
1000	2200	5	997	0	3
1000	2200	10	1000	0	0
1000	2200	15	998	0	2

Table 11: Increment 5, Temperature 2300

Number of iterations	Temperature	Initial Parameter X	Global Minimum Obtained	Other value obtained	Local Minimum Obtained
1000	2000	-15	1000	0	0
1000	2000	-10	1000	0	0
1000	2000	-5	1000	0	0
1000	2000	0	1000	0	0
1000	2000	5	1000	0	0
1000	1500	10	1000	0	0
1000	1500	15	1000	0	0

APPENDIX B MATLAB CODE 5.1

```
% simulated annealing Solution of the forth degree equation

totalglobal = 0;

totallokal=0;

totalglobalut = 0;

utenfor =0;

for o = 1:1000

k=1;

%t0 = 100;

x = 15;

x1= 15;

min = 1000;

n=1000;

b=10;

increment = 12;

maxminx = 10;

t0 = 2300;


for i=1:n

    k=k+1;
```

```
if i == 0.9*n


     increment= increment./10;

end

        a = (x^4)+2*(x^3)-(98*(x^2))+(2*x)+1;

    if min>a

 min = a;

 maxminx= x;

 x1 = x;

 else

     if i<n*0.9

     t = t0./log10(k+2);

     %abs(min)

     %abs(a)

      c = exp((-(abs((a)-min)))./t);

 d= rand(1);

    if c>d % Metropolis criteria

    x1 = x;

    end

     end
```

```
        x1=maxminx;
    end
   posneg = rand(1);

    if posneg > 0.5

     pos =  rand(1);

     x = x1 + pos*increment;

    else

        neg = rand(1);
        x = x1 - neg*increment;
    end
 end

if -7.80462< maxminx && maxminx <-7.78462
    totalglobal = totalglobal +1;
```

```
else if  maxminx>0

    totallokal = totallokal + 1;

    else if  -9 < maxminx  && maxminx < -5

            totalglobalut = totalglobalut +  1;

        else

            utenfor = utenfor +1 ;

        end


    end

end


maxminx;

end


utenfor

totalglobalut

totalglobal

totallokal
```

APPENDIX C MATLAB Code for creating the reflection Model

```
% Program creates an reflectivity series based on an zoeppritz

% approximation.



tetta=[2 10 15 17 20 30;] % Reflection for angles

rho2=[2500]; %density of layer 2

rho1=[2100];% densiy layer 1

vp1= [2800]; % VP layer 1

vp2=[3200];  % Vp layer 2

vs1=[1500]; % vs layer 1

vs2=[1800]; % Vs layer 2



Rpp = zeros(size(rho2,2),size(tetta,2));

for j = 1:size(rho2,2);


rho(j) = (rho2(j)+rho1(j))./2;

deltarho(j) = rho2(j)-rho1(j);

deltavp(j) = vp2(j)-vp1(j);

vp(j) = (vp2(j)+vp1(j))./2;
```

```
deltavs(j) = vs2(j)-vs1(j);
vs(j) = (vs2(j)+vs1(j))./2;
Zp2(j) = vp2(j)*rho2(j);
Zs2(j) = vs2(j)*rho2(j);
Zp1(j)= vp1(j)*rho1(j);
Zs1(j)= vs1(j)*rho1(j);
Zp(j) = (Zp1(j)+Zp2(j))./2;
Zs(j) = (Zs1(j)+Zs2(j))./2;

Lp(j) = 0.5*(log(Zp2(j)) - log(Zp1(j)));
Ls(j) = 0.5*(log(Zs2(j)) - log(Zs1(j)));


Rp(j) = 0.5*(deltavp(j)./vp(j) + deltarho(j)./rho(j));
Rs(j) = 0.5*(deltavs(j)./vs(j) + deltarho(j)./rho(j));
Rd(j) = (deltarho(j)./rho(j));

Rp0n(j) = (Zp2(j)-Zp1(j))./( Zp2(j) + Zp1(j)); % Normal zero offset
Rs0n(j) = (Zs2(j) - Zs1(j))./ (Zs2(j) +Zs1(j));% Normal zero offset
Rp0a = 0.5*( (deltavp(j)./vp(j)) +(deltarho(j)./rho(j))); % approximated p-wave zero offset
```

```
Rsoa = 0.5*((deltavs(j)./vs(j)) + deltarho(j)./rho(j)); % approximated s-waave
zero offset


for i = 1:size(tetta,2)


tettar(i) =tetta(i)*pi./180;


tetta;

Rpp(j,i) = (1+(tan(tettar(i)))^2)*Rp(j) -
(8*(Zs(j)./Zp(j))^2*sin(tettar(i))^2)*Rs(j) - ((0.5*tan(tettar(i))^2 -
2*(Zs(j)./Zp(j))^2*sin(tettar(i))^2))*Rd(j);



end

end


tetta;

(Rpp)



str = ['Reflection coefficients  '  num2str(Rpp)]
```

Appendix D, MATLAB code for the inversion algorithm

```
% Simulated Annealing layer model with Metropolis Criteria


reflectioncorr = [0.15362 0.14614 0.1368 0.13219 0.12448 0.09434];


for l = 1:6


   random = rand(1);
    if random > 0.5
    ru1 = rand(1)    ;
    reflectioncorr(l) = reflectioncorr(l) + 0.01*ru1;
    else
        ru2= rand(1);
       reflectioncorr(l) = reflectioncorr(l) - 0.01*ru2;
    end
 end
 for o = 1:100
 % Modeled result


tetta = [2 10 15 17 20 30]; % Reflection for angles
```

```
muzp = 700000; %Starting values

Zp = 7000000;

muzs = 4000000;

Zs = 4000000;

%murho= 0.2;

Rd = 0.2;

murd= 0.2;

%sigma = 100000;

%sigma2 = 0.1;

t0= 0.01;

%t0=sigma;

%t1=sigma;

%t2=sigma2;

k = 1;

n = 100000;

%m=0;

increment = 50000;

 incrementr = 0.01;


savedminimum(1,1) = 100;
```

```
min(1) = 10;

 for j = 1:n

 %s1= sqrt(t1);

 %s2=sqrt(t2);


 if i == n*0.9

     increment = increment/100;

 end

 Lp = 0.153942389884650; % The logarithmic reflectivities

Ls = 0.178337471969366;

 deltaz = 0;

 for i = 1:length(tetta)

tettar(i)=tetta(i)*pi./180;



Rpp = (1+(tan(tettar(i)))^2)*Lp - (8*(Zs./Zp)^2*sin(tettar(i))^2)*Ls - ((0.5*tan(tettar(i))^2 - 2*(Zs./Zp)^2*sin(tettar(i))^2))*Rd;

% reflection coefficient calculations

temp = Rpp;

 deltaz = abs(reflectioncorr(i) - temp) + deltaz;

end
```

```
if (min > deltaz )
    k = k +1;
    min = deltaz;

    savedmuzp= Zp;
    savedmuzs= Zs;
    savedmurd = Rd;
    savedminimum(1,k) = deltaz;
    muzp = Zp;
    muzs = Zs;
    murd = Rd;
else

    if j<0.9*n %Only 90 % of the runs
    t = t0/log10(j+2);
    c = exp((min-abs(deltaz))./t);
    d= rand(1);
    e =rand(1);
    f = rand(1);                    % Metropolis criteria
```

```
   if   c> d
      muzp = Zp;
   end
   if c>e
       muzs = Zs;
   end
   if c>f
       murd = Rd;
   end

  muzp=savedmuzp;
  muzs= savedmuzs;
  murd =savedmurd;
   end
  posnegr = rand(1);%Randomly picking a negative or positive number
% to make a perturbation for P-impedance
 if posnegr > 0.5
    posr =  rand(1);
    Rd = murd + posr*incrementr;
   else
```

```
          negr = rand(1);
        if  murd- negr*incrementr < 0.25


        if  murd - negr*incrementr >0
        Rd = murd - negr*incrementr;
        end
        end


    end
   posnegp = rand(1);%Randomly picking a negative or positive number
% to make a perturbation for P-impedance
     if posnegp > 0.5
        posp =  rand(1);
       Zp = muzp + posp*increment;


    else
        negp = rand(1);
        Zp = muzp - negp*increment;
    end
   posnegs = rand(1); %Randomly picking a negative or positive number
% to make a perturbation for S-impedance
```

```
    if posnegs > 0.5

     poss =  rand(1);

     Zs = muzs + poss*increment;

    else

        negs = rand(1);

        Zs = muzs - negs*increment;

    end

end

savedminimum;

 min;

 muzp;

 muzs;

 end

 totalmin(o,1) = min; % Saving the minimum value for each run
 totalmin(o,2) = savedmuzp; % Saving the pressure impedance at the minimum for each run

 totalmin(o,3)= savedmuzs;% Saving the shear impedance at the minimum for each run

 totalmin(o,4) = savedmurd;

end

totalmin

 %plot(totalmin(1:100,3),totalmin(1:100,2))
```