IDEA League

Master of Science in Applied Geophysics

Research Thesis

Moment tensor estimation through inversion of borehole microseismic data with machine learning

Eddy Dario Revelo Obando

August 8, 2021

Moment tensor estimation through inversion of borehole microseismic data with machine learning

Master of Science Thesis

for the degree of Master of Science in Applied Geophysics

by

Eddy Dario Revelo Obando

August 8, 2021

IDEA LEAGUE
JOINT MASTER'S IN APPLIED GEOPHYSICS

Delft University of Technology, The Netherlands
ETH Zürich, Switzerland
RWTH Aachen, Germany

Dated: *August 8, 2021*

Supervisor(s):

Dr. Ir. G.G. Drijkoningen
Dr. Diego Rovetta

Committee Members:

Dr. Ir. G.G. Drijkoningen

Dr. Diego Rovetta

Dr. Ir. Deyan Draganov

Dr. Cédric Schmelzbach

Abstract

The analysis of microseismic measurements acquired during borehole acquisition surveys is essential for a thorough understanding of the source mechanisms in hydraulic fracturing operations. Due to the injection of high-pressure fluids, induced fractures can produce seismic events of low magnitude that can be recorded and subsequently analyzed to infer the stress field at the location of the event.

The seismic moment tensor has been widely used to describe general seismic sources as it can provide information about the type of motion and the distribution of forces. Estimating such quantities from the recorded data can significantly improve the real-time microseismic monitoring operations and help to make assumptions about the structure of the reservoir. However, several challenges have to be faced when working with microseismic borehole measurements. They are characterized by a low signal-to-noise ratio and a limited angle coverage, which may ultimately affect the predictions about the location and the fracturing behavior inside the reservoir.

In this thesis the inversion of microseismic measurements for retrieving the moment tensor has been tackled by using a deep feedforward neural network. The seismograms used to train the network were generated through the discrete-wavenumber method. The neural network was used to predict the six independent moment-tensor components and the fault angles from seismic sources at different positions than those used during the training. The predictive capabilities of the network were tested for realistic borehole acquisition geometries and wave propagation models. The moment-tensor components were retrieved with good accuracy when using noisy seismograms and, non-double-couple source mechanisms.

As the generation of synthetic data can be expensive in terms of memory consumption, the training and prediction have been also implemented in the frequency domain using only a limited portion of the Fourier transform of the seismograms. The inversion results indicated that using narrow bands can still yield satisfactory results when predicting the moment-tensor components.

Acknowledgements

I would like to thank many people who have supported me during this period. Foremost, I would like to express my deep gratitude to my supervisor Diego Rovetta for his continuous support and patience throughout this thesis and to Guy Drijkoningen for his constructive comments and insights.

Furthermore, I would like to express my gratitude to Roald van Borselen and Diego Rovetta for allowing me to carry out my master thesis project at Aramco Overseas Company (AOC). It was a great pleasure to be part of the research center team in Delft. Many thanks to Christian Reinicke, Apostolos Kontakis, Rob Hegge, Rolf Baardman, Matteo Caporal, Paul Zwartjes, Yimin Sun, Jewoo Yoo and Marcin Dukalski for their fruitful discussions and support during my stay at the company.

Finally, I would like to thank my family for their endless support, especially my mother, father and brother. This work would not have been possible without you. Thanks for your love and for being in my life.

Delft, University of Technology
August 8, 2021

Eddy Dario Revelo Obando

Table of Contents

Chapter 1

Introduction

1-1 Microseismic monitoring of unconventional reservoirs

Throughout the last decades, the exploration of unconventional reservoirs has increased due to the development of new techniques for extracting hydrocarbons from bedrock formations. Hydraulic fracturing has become a well-known technique for extracting shale-gas oil by injecting high-pressure fluids in the subsurface to induce and propagate fractures in the surrounding rock formations (van der Baan et al., 2013). Consequently, due to the high pressure at which the fluids are being injected, microseismic events (small-scale earthquakes of low magnitude), are triggered in the adjacent areas of the reservoir. The monitoring of such events is essential in real-time production as it can help to describe the structure of the target reservoir and estimate the approximate location where the microseismic events are generated (Warpinski, 2009).

As the induced events generate seismic waves that propagate through the surrounding medium, it is possible to detect such signals using surface or downhole geophone arrays (Zhou et al., 2016). An array of geophones sense the seismic energy radiated by the source, and such information can be subsequently used to infer the orientation, extent and propagation rate of the produced fractures (Warpinski, 2009). This technique is commonly referred to as microseismic monitoring and it has become an essential tool to understand the physical processes that govern induced seismicity (Eyre and van der Baan, 2017). This approach can also be extended to other energy-related applications such as geothermal energy extraction (Häring et al., 2008), mining engineering (Sun et al., 2012), underground excavations (Collins, 2000), or CO_2 injection processes (Oye et al., 2013).

Since the events can be considered small-scale earthquakes, seismology techniques can be used to characterize the deforming mechanism caused during the fracturing of the rock (Vera Rodriguez et al., 2011). Mathematically, the seismic activity due to tectonic events or land-mass movements can be described through the seismic moment tensor (Stein and Wysession, 2005). Similarly, the seismicity induced by fluid injection in hydraulic fracturing can be described using the same mathematical approach.

1-2 Seismic moment tensor inversion

The correct estimation of the seismic moment tensor is crucial to understand the stress distributions and deforming mechanism (magnitude and direction) induced during hydraulic fracturing. This understanding leads ultimately to the effective tracking of the reservoir performance. A well-known approach for carrying out such estimation is the inversion of full-waveform data recorded from the observations at seismic receivers (Eyre and van der Baan, 2015). This has become a powerful tool also in the estimation of the physical properties of the subsurface. Hence, a great interest in finding fast and efficient algorithms to perform full-waveform inversion has surged in the exploration of (un)conventional reservoirs. The main objective is to improve the fracture treatment given a certain stress regime, fracture orientations, zones of weaknesses and pre-existing faults (Nolen-Hoeksema and Ruff, 2001).

The inversion of the seismic moment tensor was initially developed to solve the location and focal mechanisms of big-scale earthquake problems (Sipkin, 1986). Several methods have been developed to estimate such quantities from recorded data (Sambridge and Gallagher, 1993; Perol et al., 2018). Some of the most common approaches for estimating the moment tensor are described in the following sections.

1-2-1 Deterministic inversion

A physical system can be mathematically represented as a relation between direct observations carrying information about the problem of interest, unknown process or material properties, and a physical law linking the observations and unknown properties. The physical law is often represented by a mathematical function. Predicting the observations directly from the properties of the system is known as the *forward problem* (Tarantola, 2005). For instance, the Earth's gravity field can be estimated given the mass distribution of the Earth and the universal gravitational constant. In practice, however, it is desired to estimate the mass distribution from the gravity measurements, which means, the *inverse* of the forward problem needs to be solved. Finding a direct mathematical relation between the observations and the unknown physical properties is not straightforward due to the general non-linearity of real problems (Zdanov, 2015).

Estimating physical parameters in geophysics-related problems is not the exception if the complexity of the Earth's interior is considered. For example, calculating the seismic velocity from measured travel times of a wave travelling through the Earth would require solving the eikonal equation. This is obtained from the high-frequency approximation of the wave equation, where the waves are approximated as rays with propagation paths according to the Fermat's principle (Cervený, 2001). Thus, the velocity estimation is a non-linear inverse problem that can be solved using the recorded travel times at seismic stations (available data) and, the distances between two known locations (Hansen et al., 2014).

A non-linear inverse problem can be solved in a deterministic manner through the linearization of the equation linking the model parameters and the data observations. This process involves calculating the Jacobian matrix that quantifies the variation between the model parameters with respect to the observations. Then, this matrix is inverted using an initial guess of the model parameters in order to quantify the variation of the observations, and consequently, the values of the parameters.

Similarly, the seismic moment tensor can be estimated from seismic observations and the Green's functions derivatives representing the physical link between the two quantities (Song and Toksöz, 2011). Nonetheless, inverting the Green's functions to directly calculate the moment tensor is computationally expensive, and it may lead to unstable results if the problem is not properly constrained. A more detailed description of the formulation of the forward problem is given in Chapter 2.

1-2-2 Stochastic inversion

Inverse problems can also be solved following a probabilistic approach. The measurements and model parameters can be represented as random variables with a corresponding probability distribution. The latter is often assumed to be Gaussian. In stochastic approaches the inversion is carried out for multiple possible models parameters from which the one giving the best-fitting solution is chosen (Mustać and Tkalčić, 2016). By computing synthetic data, the set of parameters that yields the minimum difference between the real observations and the computed data can be chosen. This is done by means of the *likelihood* function which describes the probability of an observation under some assumptions about the model parameters.

Using independent a priori information from the model parameters, and the information from the likelihood function, it is possible to estimate the *posterior* probability density distribution of the model parameters for given observed data (Tarantola, 2005). This is achieved by means of the Bayes' theorem. Estimating the set of parameters that maximizes the posterior distribution is an optimization problem that can be solved using methods such as the Metropolis-Hastings algorithm or Simulated Annealing. Applications of seismic moment tensor inversion using stochastic approaches are described by Mustać and Tkalčić (2016) and Das et al. (2021).

1-2-3 Inversion through Artificial Neural Networks

Artificial Neural Networks (ANNs) are biologically-inspired computer programs designed to detect patterns and relationships in data (Dayhoff and DeLeo, 2001). The name Neural comes from the fact that the structure of an ANN resembles the way a biological brain collects and processes information through the nerve system in response to an external stimulus. A more detailed description on how an ANN can learn to reproduce the output of non-linear functions is given in Section 2-2-1.

Recent research activities are carried out using ANNs to solve geophysical problems such as normal moveout velocity estimation (Biswas et al., 2018, 2019), full-waveform inversion applications (Sun et al., 2020), focal-mechanisms estimation (Kuang et al., 2021), and seismic moment-tensor inversion (Ovcharenko et al., 2018). Binder (2018) used synthetic seismic wave amplitudes from microseismic events for training a feedforward neural network with one hidden layer in order to estimate the moment magnitude and the geometry of the event. Ovcharenko et al. (2018) used a deep neural network with three hidden layers to predict the full moment tensor using the amplitudes of three-component microseismic data from a single well. This approach was further extended by Carrizo Mascarell (2020) using full-waveform data from three-component seismograms to estimate the seismic moment tensor and the deforming mechanism of microseismic events.

1-3 Thesis objective and outlook

Retrieving the seismic moment tensor from borehole acquisitions is a challenging process. In general, the following factors need to be considered when working with borehole microseismic data:

1. The energy produced during hydraulic fracturing is in general weak compared to the background noise. This yields to low signal-to-noise ratio signals, affecting the performance of microseismic monitoring (Huang et al., 2017).

2. Due to the limited aperture of borehole seismic arrays for acquiring microseismic data (Vera Rodriguez et al., 2011), solving the problem in a deterministic way is not always adequate. Therefore, the correct choice of the inversion algorithm is essential to minimize the uncertainties between the real and the estimated values of the moment-tensor components.

3. Using measurements from a single well to estimate the moment tensor requires some constraints as microseismic monitoring usually needs multiple boreholes to have a better azimuthal coverage (Vavryčuk, 2007; Zhou et al., 2016).

The previous issues can be mitigated by carrying out the inversion with an ANN as suggested by many authors (Binder, 2018; Ovcharenko et al., 2018; Carrizo Mascarell, 2020). The ANN can be trained with seismograms that have been contaminated with noise, and it can yield accurate results when predicting the moment-tensor components. Additionally, by using an extensive amount of data to train the network, the issue of having a poor angle coverage can be overcome as the network can learn to recognize many combinations of angles from seismograms recorded from a single well. Furthermore, an additional advantage of using ANNs to invert the seismic moment tensor is their computational efficiency compared to deterministic or stochastic approaches.

In previous studies synthetic data generated using homogeneous velocity models (Ovcharenko et al., 2018) and source mechanisms associated with double-couple (DC) events (slip on a fault), have been used to estimate the seismic moment tensor through ANNs (Carrizo Mascarell, 2020). In this thesis, I aim to extend the previous studies to more complex scenarios, so that these techniques can be applied to field data. In particular, I have considered the following:

1. A heterogeneous velocity and density model.

2. Realistic borehole acquisition geometries used in microseismic monitoring.

3. Multiple microseismic events of variable but known position and magnitude.

4. Realistic values of signal-to-noise ratios for borehole microseismic measurements.

5. Focal mechanisms associated with shear-tensile fracturing and not pure strike-slip events.

Chapter 2 provides the mathematical definition of the seismic moment tensor, a description of the forward (how to compute full-waveform seismograms from the moment tensor) and the inverse (how to estimate the moment tensor from full-waveforms) problems, and of the ANN architecture used to predict the moment tensor from the microseismic measurements. The theory developed in Chapter 2 is used to carry out the experiments described in Chapter 3, especially for the steps needed for the generation of synthetic seismograms, training of the ANN and prediction of the moment-tensor components. In particular, the analysis of the application of the method to realistic scenarios (optimal distance of the seismic sources, effect of noise, complex source mechanisms, inversion in the frequency domain) is reported in Chapter 3. Finally, in Chapter 4 conclusions about the obtained results and some future developments are discussed.

Chapter 2

Theoretical background

One of the biggest challenges in seismology is the characterization of general seismic sources. They can be described as a series of equivalent forces producing displacements at a fixed location in space, which can be quantified from observed seismograms at the Earth's surface. The seismic moment tensor is a well-known mathematical object for describing seismic sources, as second-order tensors containing the information of the forces acting at the location where the event was produced. By inverting the recorded seismograms, the seismic moment tensor can be retrieved knowing the Earth's impulse response described by the elastodynamic Green's functions.

In this chapter, I give an overview of the mathematical derivation used to model synthetic waveforms using a known velocity and density model. I start with the description of the seismic moment tensor, its main components and its decomposition to characterize general faulting mechanisms. Afterwards, I explain how to numerically compute the Green's functions using the discrete-wavenumber method (DWM) (Bouchon and Aki, 1977) and the reflectivity method (Kennett and Kerry, 1979) for calculating the Earth's impulse response of layered media due to a general seismic source. Lastly, I describe the architecture of the ANN that is used to estimate the moment-tensor components and the preprocessing steps that are needed before inverting the data.

2-1 Forward problem

2-1-1 The seismic moment tensor

The seismic moment tensor can be thought of as the mathematical approximation of seismic sources. It can be represented as a second-order rank tensor describing the direction and magnitude of the forces acting on a determined location. Its magnitude can vary according to the strength of the event, the fault plane solution and the physical properties of the medium close to the fault's location. To start with its mathematical description, I assume that a seismic source is centered in a Cartesian coordinate system where the event is produced.

In Figure 2-1 an example of a seismic event producing fault displacement with respect to a Cartesian coordinate system is shown. The x-axis in Figure 2-1 points towards North, the y-axis towards East, and the z-axis vertically down the surface.

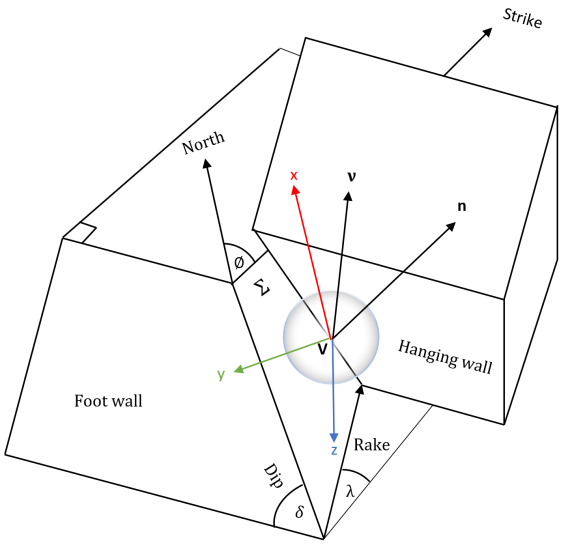


Figure 2-1. Planar fault defined by the strike ϕ and dip δ of the fault surface, and the direction λ of the slip vector (modified from Shearer (2009)).

The geometry of the fault is mainly determined by the strike, dip and rake angles represented by ϕ, δ, and λ in Figure 2-1, respectively. The vectors $\boldsymbol{\nu}$ and $\mathbf{n}$ are the slip and normal vectors to the fault. The observed seismic displacement u_n at an arbitrary position $\mathbf{x}$ and time t due to an event occurred at position $\boldsymbol{\xi}$ and time τ can be described using the so-called representation theorem (Aki and Richards, 2002), which relates the body forces and the elastodynamic Green's functions as follows:

$$u_n(\mathbf{x},t) = \int_{-\infty}^{\infty} \iiint_V G_{nk}(\mathbf{x},t;\boldsymbol{\xi},\tau) f_k(\boldsymbol{\xi},\tau)\, dV(\boldsymbol{\xi})\, d\tau, \tag{2-1}$$

where G_{nk} are the Green's functions components containing the propagation effects, f_k are the body forces acting on a source volume V, and the indices n, k identify the direction of the seismic displacement and body forces, respectively. As the forces f_k defined in equation (2-1) are not acting over the volume but on the two blocks at opposite sides of the fault plane Σ, these forces cause the foot and hanging wall to mutually move. They can be described by the *moment density tensor* $\mathbf{m}$ as follows:

$$m_{kl} = n_i \nu_j c_{ijkl}, \tag{2-2}$$

where c_{ijkl} is the stiffness tensor containing the elastic properties of the medium surrounding the fault, and i, j identify the directions of the normal and slip vectors, respectively. In addition to producing shear displacement, some microseismic events can also produce tensile fracturing. To better describe the geometry of such an event, the model shown in Figure 2-2 is considered, where the slip vector does not generally lie on the fault plane. The deviation

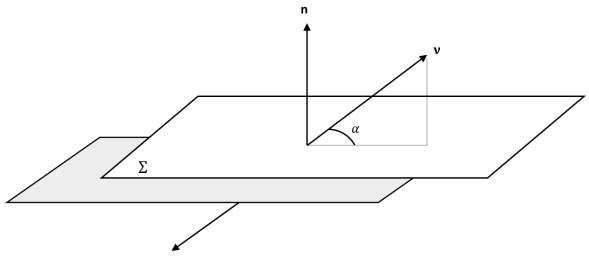


Figure 2-2. Tensile earthquake model (modified from Vavryčuk (2001)).

from the fault plane can be described by the slope angle α and the components of $\mathbf{n}$ and $\boldsymbol{\nu}$ are defined as follows:

$$\begin{aligned}
n_1 &= -\sin\delta\sin\phi,\\
n_2 &= \sin\delta\cos\phi,\\
n_3 &= -\cos\delta,\\
\nu_1 &= (\cos\lambda\cos\phi + \cos\delta\sin\lambda\sin\phi)\cos\alpha - \sin\delta\sin\phi\sin\alpha,\\
\nu_2 &= (\cos\lambda\sin\phi - \cos\delta\sin\lambda\cos\lambda)\cos\alpha + \sin\delta\cos\phi\sin\alpha,\\
\nu_3 &= -\sin\lambda\sin\delta\cos\alpha - \cos\delta\sin\alpha.
\end{aligned} \tag{2-3}$$

The magnitude of the angles determine the geometry of the deforming mechanism. Subsequently, assuming that the area over which the dipole forces are acting is of finite extent, and that the seismic signals have a long wavelength compared to the dimensions of the source (Aki and Richards, 2002; Jost and Herrmann, 1989), the point-source approximation can be used to simplify equation (2-1) as:

$$u_n(\mathbf{x},t) = \int_{-\infty}^{\infty}\iint_{\Sigma} m_{kl}(\boldsymbol{\xi},\tau)G_{nk,l}(\mathbf{x},t,\boldsymbol{\xi},\tau)\,d\Sigma\,d\tau, \tag{2-4}$$

where the forces are now represented by the moment density tensor, and l in the Green's functions term $G_{nk,l}$ describes the first spatial derivative with respect to the source coordinate in the lth direction. Moreover, the total force contribution of m_{kl} due to each infinitesimal surface $d\Sigma$ can be defined using the *seismic moment tensor* $\mathbf{M}$:

$$M_{kl} = \iint_{\Sigma} m_{kl}\,d\Sigma. \tag{2-5}$$

Equation (2-5) can be substituted in equation (2-4) to obtain an expression for the displacement in terms of the moment tensor. Due to the point-source approximation, the dependence of the moment tensor on $\boldsymbol{\xi}$ disappears (Vavryčuk, 2015), thus equation (2-4) can be written as the temporal convolution between the Green's functions derivatives and the seismic moment tensor:

$$u_n(\mathbf{x},t) = \int_{-\infty}^{\infty} M_{kl}(\tau)\,G_{nk,l}(\mathbf{x},t-\tau)\,d\tau = M_{kl}(t) * G_{nk,l}(\mathbf{x},t). \tag{2-6}$$

In equation (2-6) it is shown the relation between the recorded seismic signal, the Earth's impulse response, given by the Green's functions derivatives, and the source strength represented by the seismic moment tensor. The latter provides all the information about the

source strength and orientation of the fault plane. As the source mechanisms can vary depending on the fault type (e.g., thrust, normal, strike-slip), the magnitude of the event, and the properties of the medium, it is useful to decompose the moment tensor into different parts to better analyze the forces acting over the area of investigation.

Moment tensor decomposition

As a second-rank tensor, the seismic moment tensor can be mathematically described by a 3×3 matrix containing 9 generalized force couples (Eyre and van der Baan, 2015). The number of independent components can be further reduced due to the conservation of the angular momentum and by assuming that the medium is isotropic. The components of the seismic moment tensor can be expressed as follows:

$$\mathbf{M} = \begin{bmatrix} M_{xx} & M_{xy} & M_{xz} \\ M_{yx} & M_{yy} & M_{yz} \\ M_{zx} & M_{zy} & M_{zz} \end{bmatrix}, \tag{2-7}$$

where the subindices in equation (2-7) represent the direction of the force couple, and $M_{xy} = M_{yx}$, $M_{xz} = M_{zx}$, $M_{yz} = M_{zy}$, yielding a total of six independent values. The moment tensor can also be expressed into different components through an eigenvalue decomposition to characterize the deforming mechanism occurring at the fault's location. Using the eigenvalues m_i and eigenvectors $\mathbf{a} = [a_{ix}, a_{iy}, a_{iz}]^T$ of $\mathbf{M}$, the moment tensor can be written as follows:

$$\begin{aligned} \mathbf{M} &= \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 \end{bmatrix} \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \mathbf{a}_3^T \end{bmatrix} \\ &= \begin{bmatrix} a_{1x} & a_{2x} & a_{3x} \\ a_{1y} & a_{2y} & a_{3y} \\ a_{1z} & a_{2z} & a_{3z} \end{bmatrix} \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{bmatrix} a_{1x} & a_{1y} & a_{1z} \\ a_{2x} & a_{2y} & a_{2z} \\ a_{3x} & a_{3y} & a_{3z} \end{bmatrix}. \end{aligned} \tag{2-8}$$

The eigenvalue decomposition in equation (2-8) can be thought of as the definition of a new coordinate system comprised by the eigenvectors of the seismic moment tensor, where the source is now described as a linear combination of orthogonal dipole forces (Jost and Herrmann, 1989). Using the obtained eigenvalues ($m_1 \geq m_2 \geq m_3$), the moment tensor can be expressed into three types of sources: isotropic (ISO), double-couple (DC), and compensated-linear vector dipole (CLVD):

$$\mathbf{M} = \mathbf{M}_{ISO} + \mathbf{M}_{DC} + \mathbf{M}_{CLVD} = \mathbf{E}_{ISO} M_{ISO} + \mathbf{E}_{DC} M_{DC} + \mathbf{E}_{CLVD} M_{CLVD}, \tag{2-9}$$

where $\mathbf{E}_{ISO}$, $\mathbf{E}_{DC}$, and $\mathbf{E}_{CLVD}$ are the elementary tensors of the ISO, DC, and CLVD parts of the moment tensor:

$$\begin{aligned} \mathbf{E}_{ISO} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, & \mathbf{E}_{DC} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \\ \mathbf{E}^{+}_{CLVD} &= \frac{1}{2} \begin{pmatrix} 2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, & \mathbf{E}^{-}_{CLVD} &= \frac{1}{2} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \end{aligned} \tag{2-10}$$

and M_{ISO}, M_{DC}, and M_{CLVD} are the respective moment magnitudes:

$$\begin{aligned} M_{ISO} &= \frac{1}{3}(m_1 + m_2 + m_3), \\ M_{CLVD} &= \frac{2}{3}(m_1 + m_3 - 2m_2), \\ M_{DC} &= \frac{1}{2}(m_1 - m_3 - |m_1 + m_3 - 2m_2|). \end{aligned} \tag{2-11}$$

The isotropic part is important to quantify volume changes around the source (explosions or implosions) and it can be directly calculated from the trace of $\mathbf{M}$. The CLVD part is mainly associated with tensile and shear-tensile faulting, where the sign of the eigenvalues determines whether the deforming mechanism is compressive or extensive (Vavryčuk, 2015). Lastly, the double-couple part describes forces that produce shear faulting on a fault plane, which, in practice, is the dominant mechanism in microseismic sources (Li et al., 2021).

The moment-tensor component that does not produce shear displacement is often referred to as the non-double-couple (non-DC) or *deviatoric* part of the moment tensor, and it is the result of the sum of the ISO and CLVD parts. Moment tensors with high non-DC components may result from complex shear faulting, opening (or closing) of tensile cracks or due to anisotropy in the medium (Li et al., 2021). In this thesis the medium is assumed to be isotropic, which facilitates the calculation of the moment-tensor components by reducing the number of unknown elastic parameters. Hence, for a point source, the seismic moment tensor can be expressed as:

$$M_{kl} = \lambda_l n_i \nu_i \delta_{kl} + \mu(n_k \nu_j + n_j \nu_k), \tag{2-12}$$

where δ_{kl} is the Kronecker delta, λ_l and μ are the Lamé parameters (Vavryčuk, 2001). Under these assumptions, the number of unknown elastic parameters of c_{ijkl} is considerably reduced from 81 for a general anisotropic medium to 2. Furthermore, a general expression for the seismic moment tensor can be defined in terms of the *source tensor* $\mathbf{D}$, which includes contributions from shear and tensile fracturing:

$$M_{ij} = \lambda_l Tr(\mathbf{D})\delta_{ij} + 2\mu D_{ij}, \tag{2-13}$$

with

$$\begin{aligned} \mathbf{D} &= \frac{\bar{u}A}{2}(\mathbf{n}\boldsymbol{\nu} + \boldsymbol{\nu}\mathbf{n}) \\ &= \frac{\bar{u}A}{2}\begin{pmatrix} 2n_1\nu_1 & n_1\nu_2 + n_2\nu_1 & n_1\nu_3 + n_3\nu_1 \\ n_1\nu_2 + n_2\nu_1 & 2n_2\nu_2 & n_2\nu_3 + n_3\nu_2 \\ n_1\nu_3 + n_3\nu_1 & n_2\nu_3 + n_3\nu_2 & 2n_3\nu_3 \end{pmatrix}. \end{aligned} \tag{2-14}$$

$Tr(\mathbf{D})$ is the trace of the source tensor, $\bar{u}$ is the mean displacement on the fault plane in meters, and A is the fault area. If the slope angle in equation (2-3) is equal to 0°, then the trace of the source tensor is equal to zero and consequently the remaining force contributions of the moment tensor are due to shear faulting. This means that the double-couple part is the only non-zero part of the moment tensor.

Dahm and Krüger (2014) describe a different approach for characterizing source mechanisms having shear and tensile motion along the fault plane, where the six independent moment-tensor components can be written as follows:

$$
\begin{aligned}
M_{xx} &= -D_s A\mu(\sin 2\phi \sin\delta \cos\lambda + \sin^2\phi \sin 2\delta \sin\lambda) + D_n A(\lambda_l + 2\mu \sin^2\phi \sin^2\delta),\\
M_{xy} &= D_s A\mu(\cos 2\phi \sin\delta \cos\lambda + 0.5 \sin 2\phi \sin 2\delta \sin\lambda) - D_n A\mu \sin 2\phi \sin^2\delta,\\
M_{xz} &= -D_s A\mu(\cos\phi \cos\delta \cos\lambda + \sin\phi \cos 2\delta \sin\lambda) + D_n A\mu \sin\phi \sin 2\delta,\\
M_{yy} &= D_s A\mu(\sin 2\phi \sin\delta \cos\lambda - \cos^2\phi \sin 2\delta \sin\lambda) + D_n A(\lambda_l + 2\mu \cos^2\phi \sin^2\delta),\\
M_{yz} &= -D_s A(\sin\phi \cos\delta \cos\lambda - \cos\phi \cos 2\delta \sin\lambda) - D_n A\mu \cos\phi \sin 2\delta,\\
M_{zz} &= D_s A\mu \sin 2\delta \sin\lambda + D_n A(\lambda_l + 2\mu \cos^2\delta),
\end{aligned}
\tag{2-15}
$$

where D_s and D_n denote the magnitude of the shear displacement and normal displacement to the fault plane, respectively. In equation (2-15), the moment-tensor components have contributions from both types of motions. Seismic sources producing shear and tensile motions are often referred to as *general dislocation sources* (Li et al., 2021). Figures 2-3a and 2-3b show a better description of a shear and tensile crack over the fault plane Σ.

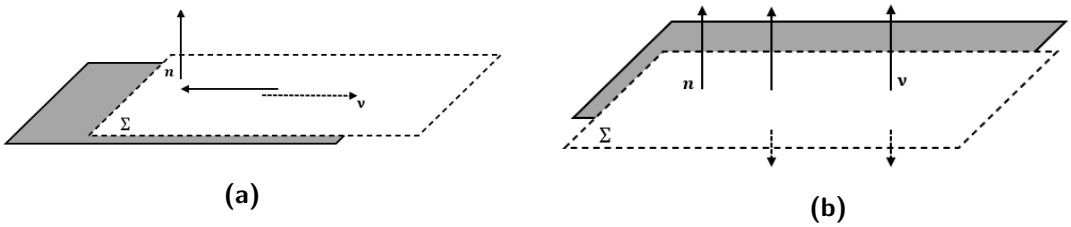


Figure 2-3. General dislocation source: (a), motion parallel to the fault plane; (b), motion perpendicular to the fault plane (modified from Dahm and Krüger (2014)).

In Figure 2-3a, the slip vector lies on the fault plane and forms an angle of 90° with the normal vector. For a pure tensile crack (Figure 2-3b), the normal and slip vectors are parallel to each other and no shear motion is produced. If the slip vector deviates from the fault plane such that the angle formed with the normal vector is neither 0° nor 90°, the resulting displacement has contributions from both types of motions.

Furthermore, equation (2-15) can also be derived from equation (2-13). When α in Figure 2-2 is 0°, the resulting movement is pure shear as the first term on the right-hand side of equation (2-13) cancels out. Similarly, if α is equal to 90°, the movement is purely extensive (or compressive), and the second term on the right-hand side of equation (2-13) has only non-negative values on the diagonal of $\mathbf{D}$ due to the ISO part of the moment tensor. Likewise, when the magnitude of D_n is 0 m, the expressions for the moment-tensor components in equation (2-15) are equivalent to equation (2-13) if the angle between the slip and normal vectors is 0°.

The magnitude of the shear displacement can be estimated theoretically knowing the magnitude of the event, the approximated area of the fracture and its rupture length. As the previous quantities depend on the many factors involved in the fracking operation e.g., the physical properties of the surrounding rocks, estimating the exact area of the induced fracture

is not straightforward as it can have an irregular shape. In this thesis, the area of the fracture is assumed to be approximately circular with a radius equal to half of the value of the rupture length. Shapiro (2015) gives an approximate relation from which the rupture length R can be calculated given a known *moment magnitude* M_w:

$$M_w \approx 2\log_{10}(R) - 1. \tag{2-16}$$

The moment magnitude is a dimensionless parameter used to quantify the strength of a seismic event. By calculating the value of the rupture length using equation (2-16), an approximate area of the fracture can be obtained. Further, to estimate the shear displacement from the previous quantities, a relation between the moment magnitude and the produced motion needs to be defined. Thus, it is more convenient to express M_w in terms of the *scalar seismic moment* M_0:

$$M_w = \frac{2}{3}(\log_{10}(M_0 - 9.1)), \tag{2-17}$$

where the units of M_0 are given in Newtons per meter (Aki and Richards, 2002; Kanamori, 1977). By inverting equation (2-17) to calculate M_0, the shear displacement can be solved using the following relation:

$$M_0 = \mu D_s A. \tag{2-18}$$

The value of D_s is commonly given in meters for big-medium scale earthquakes with moment magnitudes M_w ranging from 4 to 6, and a few millimeters for small events with negative moment magnitudes (McGarr, 2003).

2-1-2 Modified Discrete Wavenumber method

Having defined the seismic moment tensor for representing general seismic sources, the next step corresponds to the calculation of the elastodynamic Green's functions that are used to generate synthetic waveforms using a known source mechanism, velocity and density model. The Green's functions are calculated through the discrete-wavenumber method (DWM) described in Bouchon and Aki (1977).

The seismic wavefield can be represented by an elastic isotropic point source in a cylindrical coordinate system (r, z) where r is the radius, and z the depth coordinate. The potential of such source can be expressed as an integral over the radial wavenumber (in the horizontal plane) domain as follows:

$$\phi(r, z; \omega) = \frac{iQ_s(\omega)}{4\pi} \int_0^\infty \frac{k_r}{k_{z,p}} J_0(k_r r) e^{-ik_{z,p}|z|}\, dk_r, \tag{2-19}$$

where $Q_s(\omega)$ represents the volume change of the source, ω is the angular frequency, k_r is the radial wavenumber, J_0 denotes the zeroth-order Bessel function, and $k_{z,p}$ is the P-wave vertical wavenumber defined by:

$$k_{z,p} = \sqrt{\frac{\omega^2}{c_p^2} - k_r^2}, \quad \mathrm{Im}(k_{z,p}) < 0, \tag{2-20}$$

where c_p is the compressional wave velocity. The aim of the DWM method is to express an elastic point source as the sum of N plane waves distributed horizontally at equal intervals Δr to facilitate the integration over the radial wavenumber (see Figure 2-4). The discretization of the expression for the radiated wavefield can then be expressed as follows:

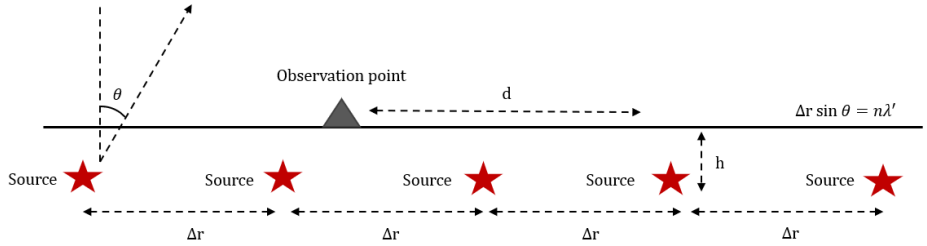


Figure 2-4. Interpretation of the DWM. An elastic point source is expressed as a finite array of sources horizontally distributed at intervals Δr, d is the horizontal distance from the observation point to the source, h is the vertical distance from the surface to the source position, and $\lambda^{'}$ is the wavelength of the radiated wavefield (modified from Bouchon (2003)).

$$\phi(r,z;\omega) = \frac{iQ_s(\omega)}{2}\sum_{n=0}^{N}\frac{k_{nr}}{k_{nz,p}}J_0(k_{nr}r)e^{-ik_{nz,p}|z|}, \tag{2-21}$$

with

$$k_{nr} = \frac{2\pi}{\Delta r}n, \quad k_{nz,p} = \sqrt{\frac{\omega^2}{c_p^2} - k_{nr}^2}. \tag{2-22}$$

The summation over all the wavenumbers in equation (2-21) yields the approximation of the total wavefield in equation (2-19). This series is truncated by calculating the contribution of each wavenumber for every frequency ω with respect to the current sum. The calculation can be stopped if adding more wavenumbers to the series does not yield an improvement in the estimated potential. The discretization of the DWM can also be used to express the wavefield radiation of a source of components $\mathbf{F} = (F_x, F_y, F_z)$. In a cylindrical coordinate system (r, θ, z) centered at the source position, the compressional and rotational potentials Φ, Ψ, and χ for the P, SV, and SH waves, respectively, are defined as follows:

$$\begin{aligned}\Phi(r,\theta,z;\omega) = \frac{1}{2\Delta r\rho\omega^2}\Bigg[&\mathrm{sgn}(z)F_z\sum_{n=0}^{N}k_{nr}J_0(k_{nr}r)e^{-ik_{nz,p}|z|} \\ &- i(F_x\cos\theta + F_y\sin\theta)\sum_{n=0}^{N}\frac{k_{nr}^2}{k_{nz,p}}J_1(k_{nr}r)e^{-ik_{nz,p}|z|}\Bigg],\end{aligned} \tag{2-23}$$

$$\Psi(r,\theta,z;\omega) = \frac{1}{2\Delta r\rho\omega^2}\Bigg[- iF_z \sum_{n=0}^{N} \frac{k_{nr}}{k_{nz,s}} J_0(k_{nr}r) e^{-ik_{nz,s}|z|} + \operatorname{sgn}(z)(F_x\cos\theta + F_y\sin\theta) \sum_{n=0}^{N} J_1(k_{nr}r) e^{-ik_{nz,s}|z|}\Bigg], \tag{2-24}$$

$$\chi(r,\theta,z;\omega) = i\frac{F_y\cos\theta - F_x\sin\theta}{2\Delta r\rho c_s^2} \sum_{n=0}^{N} \frac{1}{k_{nz,s}} J_1(k_{nr}r) e^{-ik_{nz,s}|z|}, \tag{2-25}$$

with

$$k_{nz,s} = \sqrt{\frac{\omega^2}{c_s^2} - k_{nr}^2}, \quad \operatorname{Im}(k_{nz,s}) < 0, \tag{2-26}$$

and

$$\operatorname{sgn}(z) = \begin{cases} 1, & \text{for } z > 0 \\ -1, & \text{for } z < 0 \end{cases}, \tag{2-27}$$

where J_1 is the Bessel function of first order, ρ is the density of the medium, and c_s is the shear-wave velocity. Making use of the Helmholtz theorem as in Cotton and Coutant (1997), the Green's functions can be computed by differentiating the respective wavefield potentials:

$$\mathbf{G}(r,\theta,z) = \nabla\Phi + \nabla\times(\mathbf{e}_z\Psi) + \nabla\times\nabla\times(\mathbf{e}_z\chi), \tag{2-28}$$

where $\mathbf{e}_z$ is the unit vector in the z direction.

Depending on the chosen Earth's velocity and density model, these potentials are propagated through the medium using the reflection and transmission coefficients at each interface, yielding upward and downward potentials radiated from the source layer (Kennett and Kerry, 1979). Then, the recorded wavefields at the receivers stations are used for calculating the Green's functions derivatives. This procedure is detailed in Appendix A.

2-1-3 Forward modeling algorithm

According to Aki and Richards (2002), the elastic radiation from a seismic source can be obtained by the superposition of elementary point sources. Thus, it is necessary to modify the DWM described in Section 2-1-2 to account for the effects of a more complex source. The approach used by Cotton and Coutant (1997) considers a source mechanism composed by six elementary moment tensors M^i: one isotropic and five double-couples, each one multiplied by a trigonometric coefficient a_n:

$$\mathbf{M^1} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \qquad \mathbf{M^2} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix},$$
$$\mathbf{M^3} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}, \quad \mathbf{M^4} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{2-29}$$
$$\mathbf{M^5} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad \mathbf{M^6} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

The full moment tensor can be obtained as a linear combination of the elementary tensors as:

$$\mathbf{M} = \sum_{n=1}^{6} a_n \mathbf{M}^n = \begin{bmatrix} -a_4 + a_6 & a_1 & a_2 \\ a_1 & -a_5 + a_6 & -a_3 \\ a_2 & -a_3 & a_4 + a_5 + a_6 \end{bmatrix}, \tag{2-30}$$

and

$$\begin{aligned} a_1 &= M_{xy}, \\ a_2 &= M_{xz}, \\ a_3 &= -M_{yz}, \\ a_4 &= \frac{-2M_{xx} + M_{yy} + M_{zz}}{3}, \\ a_5 &= \frac{M_{xx} - 2M_{yy} + M_{zz}}{3}, \\ a_6 &= \frac{M_{xx} + M_{yy} + M_{zz}}{3}. \end{aligned} \tag{2-31}$$

The coefficients a_n are determined by the moment-tensor components estimated after recording the different source potentials at the receiver stations (Cotton and Coutant, 1997). Assuming that the elementary tensors have the same time dependence, and using the relation defined in equation (2-6), six elementary seismograms E_i^n can be calculated by convolving M^i with the Green's functions derivatives:

$$E_i^n(\mathbf{x}, t) = \sum_{k,l} G_{ikl}(\mathbf{x}, t) * M_{kl}^n(t), \tag{2-32}$$

where the superscript n represents the associated elementary tensor. The seismograms of equation (2-6) can therefore be expressed as:

$$u_i(\mathbf{x}, t) = \sum_{n=1}^{6} E_i^n(\mathbf{x}, t) a_n. \tag{2-33}$$

The previous equation is valid when the source wavelet has an infinite band (Dirac's distribution). If a general source wavelet $S(t)$ is taken into account, then the previous expression becomes:

$$u_i^{'}(\mathbf{x},t) = \left(\sum_{n=1}^{6} E_i^n(\mathbf{x},t) a_n \right) * S(t). \tag{2-34}$$

Similarly, the seismograms can be computed in the frequency domain to further simplify the calculations as:

$$\hat{u}_i^{'}(\mathbf{x},\omega) = \left(\sum_{n=1}^{6} \hat{E}_i^n(\mathbf{x},\omega) a_n \right) \hat{S}(\omega), \tag{2-35}$$

where $\hat{u}_i^{'}(\mathbf{x},\omega)$, $\hat{E}_i^n(\mathbf{x},\omega)$, and $\hat{S}(\omega)$ are the Fourier transformations of the seismic displacement, the elementary seismograms, and the source wavelet, respectively.

2-2 Inverse problem

In Section 2-1-3 it was described how to calculate synthetic waveforms knowing the Green's functions derivatives and the moment-tensor components. In this section, I analyze the problem of retrieving the seismic moment tensor directly from the recorded seismograms. The seismic moment tensor defined in equation (2-6) can be expressed in the frequency domain as follows:

$$\hat{u}(\mathbf{x},\omega) = \hat{M}_{kl}(\omega) \hat{G}_{nk,l}(\mathbf{x},\omega), \tag{2-36}$$

where $\hat{u}(\mathbf{x},\omega)$, $\hat{M}_{kl}(\omega)$, and $\hat{G}_{nk,l}(\mathbf{x},\omega)$ are the Fourier transformations of the seismograms, the moment tensor, and the spatial derivatives of the Green's functions, respectively. The terms in $\hat{M}_{kl}(\omega)$ can be rearranged to express the six independent moment-tensor components as a vector. Hence, equation (2-36) may be rewritten as a linear system of equations as follows:

$$\mathbf{u} = \mathbf{G}\mathbf{m}, \tag{2-37}$$

where $\mathbf{G}$ has dimensions of $K \times 6$, for K seismic sensors,

$$\mathbf{G} = \begin{pmatrix} G_{11} & G_{12} & \cdots & G_{16} \\ G_{21} & G_{22} & \cdots & G_{26} \\ \cdots & \cdots & \cdots & \cdots \\ G_{K1} & G_{K2} & \cdots & G_{K6} \end{pmatrix}. \tag{2-38}$$

The moment-tensor components $\mathbf{m}$ are written in vector form,

$$\mathbf{m} = \begin{bmatrix} M_{xx} & M_{yy} & M_{zz} & M_{xy} & M_{xz} & M_{yz} \end{bmatrix}^T, \tag{2-39}$$

and the vector $\mathbf{u}$ contains the measurements of the three-component sensors,

$$\mathbf{u} = \begin{bmatrix} u_x^1 & u_y^1 & u_z^1 & u_x^2 & u_y^2 & u_z^2 & \cdots & u_x^K & u_y^K & u_z^K \end{bmatrix}^T. \tag{2-40}$$

The final seismic displacements are then expressed in the time domain via inverse Fourier transformation of each component. These calculations are performed for each source mechanism to be modeled, which may result computationally expensive depending on the dimensions

of the chosen velocity and density model, and the number of wavenumbers used to calculate the summations described in equations (2-23), (2-24), and (2-25) to compute the respective wave potentials. Equation (2-37) can be directly inverted for the moment-tensor components by calculating the pseudo-inverse of $\mathbf{G}$ as follows:

$$\mathbf{m} = \mathbf{G}^{+}\mathbf{u} = (\mathbf{G}^{T}\mathbf{G})^{-1}\mathbf{G}^{T}\,\mathbf{u}. \tag{2-41}$$

Calculating $\mathbf{G}^{+}$ is however complicated and computationally expensive. Therefore, the inversion of the moment-tensor components can be carried out using ANNs so that the computational effort is moved before the training phase when the seismograms are pre-calculated for a given location before being fed to the network. However, implementing a suitable network architecture is not a straightforward process since it requires a thorough analysis for choosing the best parameters that lead to the highest prediction accuracy. Recent developments of machine learning frameworks such as TensorFlow and Keras, allow an efficient implementation of supervised learning techniques.

2-2-1 ANN achitecture design

Artificial Neural Networks made their first appearance in McCulloch and Pitts (1943), where the authors described a computational model of the neural activity in animal brains, and how they might carry out complex calculations. Their proposal consisted of modelling biological neurons as artificial units performing binary operations, where the number of active inputs at the neuron determines the activation of the output. They demonstrated that the model of an artificial neuron could be extended to build a simple network that performs logical computations, e.g., logical gates such as OR, AND or NOR.

This idea was further developed when Rosenblatt (1958) introduced the concept of the *Perceptron* to model an artificial neuron. A neuron's output $\boldsymbol{y}$ can be described as the weighted sum of the data observations $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ and a set of parameters $\boldsymbol{w} = (w_1, w_2, \cdots, w_n)$ often referred to as *weights*. The weighted sum is then passed through an activation function $f(\cdot)$ that determines the value of the activation threshold of the neuron (see Figure 2-5).

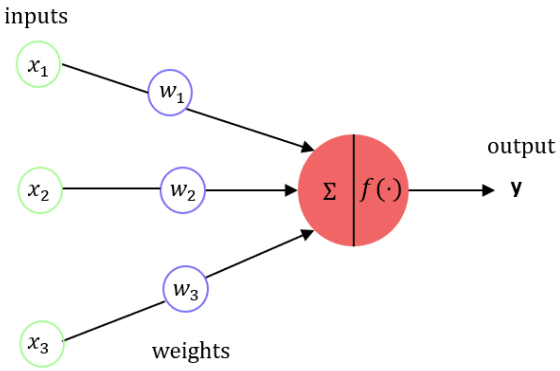


Figure 2-5. Model of an artificial neuron.

A basic feedforward architecture is constructed using an ensemble of artificial neurons organized in *layers*. Each unit (neuron) in every layer is connected to every unit in the next layer,

except for the output layer. An example of an ANN is shown in Figure 2-6. The data are fed to the input layer and subsequently propagated through the *hidden* layers of the network up to the output layer. The optimal result is obtained by iteratively updating the weights so that the difference between the input and the calculated output is reduced. This process is referred to as *training* since the neural network is learning to recognize an output given a specific input. In other words, it is possible to predict an output from an input not used during the training. The number of times that a dataset needs to pass through the network to achieve acceptable results is called an *epoch*. A neural network needs a number of epochs to train the data which depends on the application and on the input data. A detailed description of the training process is given in Section 2-2-2.

In this thesis a deep feedforward architecture (Figure 2-6) is proposed to carry out the moment tensor inversion using three-component seismograms as input. The set of parameters that needs to be defined for implementing the architecture of an ANN are referred to as *hyperparameters* because they need to be set manually, and are not estimated during the training. The main hyperparameters that define the network's architecture are listed as follows:

- Number of hidden layers.
- Number of neurons per layer.
- Activation functions.

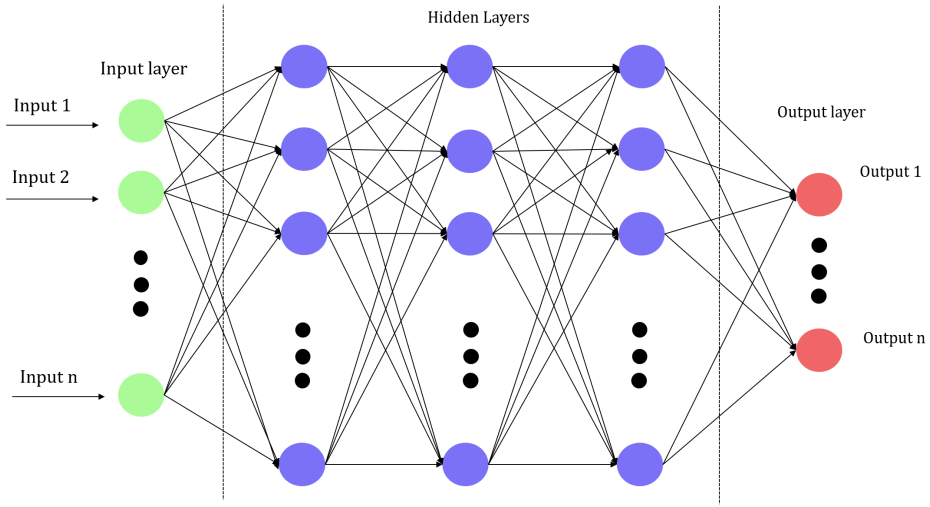


Figure 2-6. Example of a deep feedforward neural network with three hidden layers (modified from Tano et al. (2020)).

The depth of an ANN is determined by the number of its hidden layers. Hence, deep neural networks are architectures having more than one hidden layer. After passing the data through the input layer, every unit in the hidden layers performs a non-linear operation using the inputs from the previous layer. This non-linear mapping is determined by the activation function, allowing the neural network to learn complex functions.

ANNs have become very popular in many scientific and commercial applications such as object recognition, image classification or natural language processing. Some of these applications

require more complex architectures e.g., convolutional or recurrent neural networks. The ANN architecture proposed in this thesis is based on the previous works carried out by Ovcharenko et al. (2018) and Carrizo Mascarell (2020) on moment tensor inversion. Their work has been extended in order to be applied to more realistic scenarios and ultimately to field data. The number of hidden layers, units, and activation functions per layer of the proposed ANN are summarized in Table 2-1.

Table 2-1. Artificial Neural Network architecture.

Layer	Number of units	Activation function	# Parameters
Input layer	$3 \times N_r \times N_s$	-	$3 \times N_r \times N_s$
1^{st} hidden layer	164	$\tanh(\cdot)$	$3 \times N_r \times N_s \times 164$
2^{nd} hidden layer	92	$\tanh(\cdot)$	164×92
3^{rd} hidden layer	64	$\tanh(\cdot)$	92×64
Output layer	6	-	64×6

In the first column the type of layer is described. In the second, the number of units per layer, where N_r represents the number of receiver stations, N_s the number of time samples per seismogram and the number 3 accounts for the three components of the seismic sensor. The activation function of the hidden layers is the hyperbolic tangent $\tanh(\cdot)$. The output layer where the six independent moment-tensor components are obtained does not contain any activation function. The dimensions of the matrices containing the network's parameters at each layer depend on the number of neurons of the current and previous layers (see Table 2-1), except for the input layer where it only depends on the size of the training data.

2-2-2 ANN training and validation

Data normalization

A crucial step before starting the training phase is data normalization. To facilitate the training process, the input data that is fed to the network needs to have the same range of values to simplify the learning process. The normalization approach used in this thesis is data standarization, where for every element x_i of the input $\boldsymbol{x}$, the following transformation is applied:

$$x_s = \frac{x_i - \bar{x}}{\sqrt{\frac{\Sigma_i (x_i - \bar{x})^2}{N}}}, \tag{2-42}$$

where x_s is the transformed element of x_i, N is the size of the input, $\bar{x}$ is the mean of the input $\boldsymbol{x}$, and at the denominator the standard deviation of the input is computed.

Data training and validation sets

It is essential that the trained network can generalize well on data that were not previously used during the training phase. Thus, the dataset needs to be split into three different subsets:

training set, validation (or hold-out) set, and test set. The training set is used to fit the model and estimate the values of $\boldsymbol{y}$ from the inputs $\boldsymbol{x}$. The validation set is used to keep track of the error between the estimated and true values at every epoch with the purpose of avoiding *overfitting* in the model. This occurs when the gap between the validation and the training error is too large (Bishop, 2006). In this situation, the network is not able to accurately predict data that were not used during the training.

Optimization

This section aims to describe the training process of a neural network to compute the optimal weights $\boldsymbol{w}$ that reduce the error between the input and the predicted output. To calculate such weights, a proper optimization algorithm needs to be chosen.

Let's consider an input vector $\boldsymbol{x}$ that is fed to a hidden neuron in the network. At the input of the neuron, an element-wise non-linear transformation is calculated through the activation function as follows:

$$h^{(i)}(\boldsymbol{x}; w^{(i)}, b^{(i)}) = f(w^{(i)T}\boldsymbol{x} + b^{(i)}), \tag{2-43}$$

where $h^{(i)}$ is the result of the non-linear transformation at the i^{th} layer for every neuron, and $w^{(i)}$ and $b^{(i)}$ are the network's weights and bias parameters to be optimized. For practical considerations, a new vector $\theta^{(i)} = [b^{(i)}\, w^{(i)}]^T$ is defined to simplify the matrix operations performed in equation (2-43). After feeding the data at the input layer, the network propagates it up to the hidden layers to produce the estimated values $\hat{\boldsymbol{y}}$ for the moment-tensor components. At the output layer, the value of a scalar loss function $J(\boldsymbol{\theta})$ is calculated to measure the error between the true and estimated values. This function can be written as the mean-squared-error between these quantities as follows:

$$J(\boldsymbol{\theta}) = \frac{1}{N}\sum_{n=1}^{N} ||\hat{y}_n(x_n; \boldsymbol{\theta}) - y_n||^2, \tag{2-44}$$

where y_n are the target values, and N the number of inputs. To calculate the optimal set of parameters $\boldsymbol{\theta}$, the loss function defined in equation (2-44) needs to be minimized. This function can be minimized with a gradient-descent algorithm, where the computed gradient measures the changes of the loss function in the direction of maximum change. The parameters $\boldsymbol{\theta}$ can be updated using the following scheme:

$$\theta^{(i+1)} = \theta^{(i)} - \eta \nabla J(\theta^{(i)}), \tag{2-45}$$

where $\theta^{(i+1)}$ are the updated parameters at the current epoch, $\theta^{(i)}$ are the parameters of the previous epoch, η is the learning rate, and $\nabla J(\boldsymbol{\theta})$ is the gradient of the loss function. Estimating the gradient to update the parameters of the ANN would require calculating partial derivatives of the loss function with respect to $\boldsymbol{\theta}$ at every layer. Such operations can be carried out through the *backpropagation* algorithm, where all the derivatives are computed using the chain rule starting with the loss function at the output layer. Subsequently, using the resulting derivatives, backpropagation is performed until the parameters are updated.

Furthermore, the use of activations functions such as the logistic sigmoid and hyperbolic tangent facilitates the calculation of the gradient throughout the network because their derivatives can be expressed in terms of the function itself. For instance, the first derivative of the

hyperbolic tangent function with respect to a parameter z is equal to:

$$\frac{d}{dz}\tanh(z) = 1 - \tanh^2(z), \tag{2-46}$$

where there is no need of applying any approximation for the derivative. After calculating the gradient of the loss function, the model parameters in equation (2-45) need to be updated using the values of $\boldsymbol{\theta}$ from the previous epoch and an appropriate learning rate. Setting the right value for η is however a big challenge in optimization theory because it can significantly affect the learning process as it determines the size of the step taken in the negative direction of the gradient. Many of the optimization algorithms used in modern machine learning applications are based on the gradient-descent method, therefore, new variations have been developed in last years that can mitigate this issue (Goodfellow et al., 2016).

Based on the work carried out by Carrizo Mascarell (2020), the *RMSprop* optimization algorithm is chosen for minimizing the loss function. The big advantage compared to the gradient-descent method is that the learning rate is iteratively adapted to correct for the direction of the gradient toward the global minimum. The gradient at the current epoch is scaled with the square root of the sum of the squared values of the gradient from the previous epoch. Consequently, the learning rate decays faster for parameters with large partial derivatives of the loss function, and it decays slower for parameters with small partial derivatives (Goodfellow et al., 2016).

However, if the gradient vector is normalized using the sum of the squared values of the gradient from the beginning of the training, the algorithm may end up converging earlier without having reached the global minimum. This is because of the rapid decrease of the learning rate. For example, the *AdaGrad* algorithm (Duchi et al., 2011) suffers from this issue, which makes it not suitable for many deep learning applications. RMSprop corrects for this problem by introducing an exponentially decaying average to the sum of the squared values of the gradient at the first iteration. In other words, the algorithm accumulates the values from the most recent iterations yielding a big improvement in the training process (Goodfellow et al., 2016). RMSprop main equations are:

$$1. \quad s^{(i)} = \beta s^{(i-1)} + (1-\beta)\nabla J(\theta^{(i)})\nabla J(\theta^{(i)})^T, \tag{2-47a}$$

$$2. \quad \theta^{(i+1)} = \theta^{(i)} - \frac{\eta}{\sqrt{s^{(i)} + \epsilon}}\nabla J(\theta^{(i)}), \tag{2-47b}$$

where $\mathbf{s}$ denotes the cumulative sum of the squared values of the gradient at i^{th} epoch, β controls the length scale of the moving average of the gradient-descent values, and ϵ is a regularization term to avoid division by zero in equation (2-47b). In step 1, the sum of the squares of the gradient is calculated, and the resulting value is used to update the network's parameters using a different learning rate for every epoch. β and ϵ are also hyperparameters that need to be set manually when defining the optimizer of the network. In practice, they do not need to be tuned as default values usually work well.

Training the ANN with RMSprop using the entire dataset becomes computationally expensive and not efficient when the size of the input is large. Therefore, it is convenient to train the network using a small subset of the total number of available samples (Goodfellow et al., 2016). The size of such subset is chosen to range from ten to a few hundred samples and is then used to evaluate the loss function and update the network parameters $\boldsymbol{\theta}$. The optimization

methods that use a portion of the dataset to estimate the gradient are referred to as *mini-batch* methods because a small "batch" or "mini-batch" of the data is used to train the network. In this thesis, a combination of RMSprop with mini-batches is used for training the network as it results to be an efficient strategy that yields an acceptable outcome.

2-2-3 ANN prediction

After ensuring that the trained network does not overfit the data having an acceptable validation error, the network can be tested on data not used during the training and validation. To measure how well non-analyzed data can be predicted using the trained network, the coefficient R^2 is used:

$$R^2(\boldsymbol{y}, \hat{\boldsymbol{y}}) = 1 - \frac{\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}{\sum_{n=1}^{N}(y_n - \bar{y})^2}, \quad \bar{y} = \frac{1}{N}\sum_{n=1}^{N} y_n, \tag{2-48}$$

where $\hat{y}_n$ and y_n are the estimated and target values, respectively (James et al., 2013). The coefficient R^2 indicates in this particular case how well the trained network fits data that was not previously used during the training. Its values range from 0 to 1, where the latter would imply that the predicted values match exactly the trained ones. In the following chapters, R^2 is referred to as the prediction accuracy of the results.

Throughout this thesis several tests are performed for predicting the moment-tensor components on different scenarios using the same set of hyperparameters. The optimization algorithm and set of parameters chosen for solving such problems are summarized in Table 2-2.

Table 2-2. Neural Network settings.

Optimizer: RMSprop	Learning rate (η) = 0.001 $\beta = 0.9$ $\epsilon = 1 \times 10^{-7}$
Batch size	256
# Epochs	100
Early stopping criterion	Tolerance = 0.001
Loss function	Mean-squared-error (MSE)

The tolerance for the early stopping criterion represents the minimum change in the validation error that can be quantified as an improvement in the learning process. The monitored quantity is the absolute value of the difference between the validation error of the loss function between the current and the previous epoch. The network stops the training if there is not a significant change in the validation loss after a determined number of epochs.

Chapter 3

Application to realistic scenarios

Estimating the seismic moment tensor from data observations can be solved in several manners. In a deterministic approach, the solution is given by the direct inversion of equation (2-37) which would require calculating the inverse matrix of $\mathbf{G}$, and then multiplying it by the vector of observations $\mathbf{d}$ to retrieve the moment-tensor components $\mathbf{m}$ as in equation (2-41).

Ovcharenko et al. (2018) and Carrizo Mascarell (2020) successfully estimated the moment-tensor components using deep feedforward neural networks. The former used first-arrivals' peak value amplitudes of the seismograms as inputs for the ANN, whereas the latter used full-waveforms generated by a single microseismic event in an elastic homogeneous medium. I aim to extend the analysis of the moment-tensor inversion using heterogeneous velocity and density models, non-fixed source positions, and non-double-couple mechanisms. The purpose is to predict the moment-tensor components and the respective angles of the fault plane for realistic borehole acquisition geometries.

3-1 Training strategy for realistic scenarios

A real borehole microseismic acquisition geometry is shown in Figure 3-1. It consists of a buried horizontal well having 20 three-component receivers with a spacing of 25 m between them covering a distance of 500 m. The microseismic events are assumed to be positioned 300-400 m aside of the receiver array in the West-East direction. The blue triangles represent the receiver stations spread in the North-South direction. The red points are the microseismic events used for training the neural network, whereas the green points are the events whose moment tensor needs to be estimated using the trained network. In other words, the seismic information generated by the events marked in red in the geometry, is used to predict the moment-tensor components for events not previously seen by the network (marked in green in the figure).

The SEAM Arid model (Oristaglio, 2015) has been chosen to simulate elastic wave propagation in a realistic scenario. The model is characterized by heterogeneous properties (velocities and densities) and features typical of desertic environments, such as layered sequences of sands

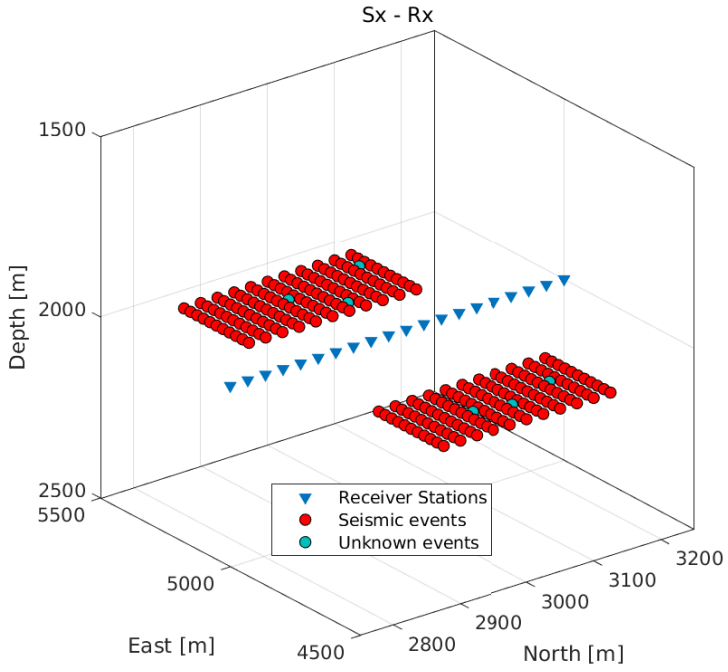


Figure 3-1. Proposed geometry setup resembling a realistic borehole seismic acquisition.

and shales and karsts. The latter are cavities in the subsurface formed by the dissolution of carbonate rocks.

In particular, a 1D profile has been extracted from the 3D model and used to generate synthetic seismograms through a 1.5D modeling code. The software has been developed in Python following the discrete-wavenumber method described in Chapter 2. The P-wave, S-wave velocities (c_p and c_s) and density (ρ) profiles are shown in Figure 3-2.

The total depth of the chosen profile is 3.75 km with 600 layers of different velocity and density values sampled every 6.25 m. Furthermore, the characteristics of the microseismic events also need to be specified. For the analysis carried out in this chapter, only events of magnitudes $M_w = -2$ are considered. This is a reasonably standard value encountered in microseismic monitoring. The fault angles (strike ϕ, dip δ, and rake λ) are also essential to define the seismic moment tensor. They range as follows:

$$\begin{aligned} &\text{Strike range} : 0^\circ \leq \phi \leq 360^\circ. \\ &\text{Dip range} : 0^\circ \leq \delta \leq 90^\circ. \\ &\text{Rake range} : -180^\circ \leq \lambda \leq 180^\circ. \end{aligned} \tag{3-1}$$

Two examples of synthetic three-component seismograms using the SEAM Arid model for two different sets of angles (ϕ, δ, λ) are depicted in figures 3-3a and 3-3b.

In both figures 3-3a and 3-3b the P-wave, S-wave arrivals and some multiple reflections can be seen from the obtained seismograms. Depending on the orientation of the fault plane, the

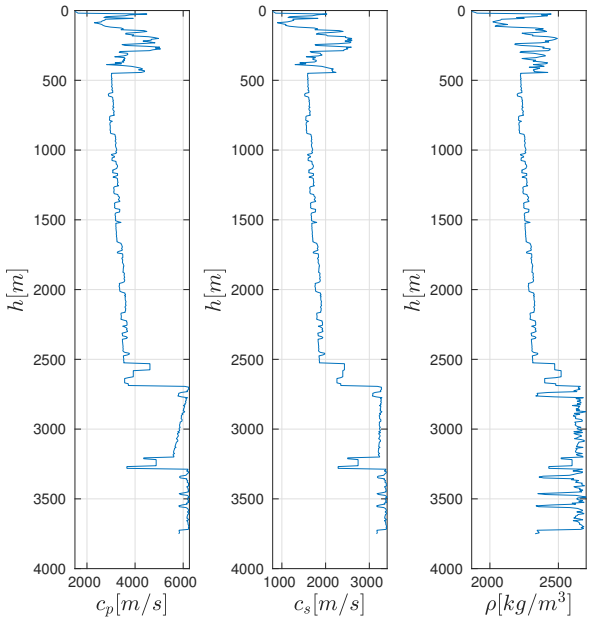


Figure 3-2. 1D profile - SEAM Arid model.

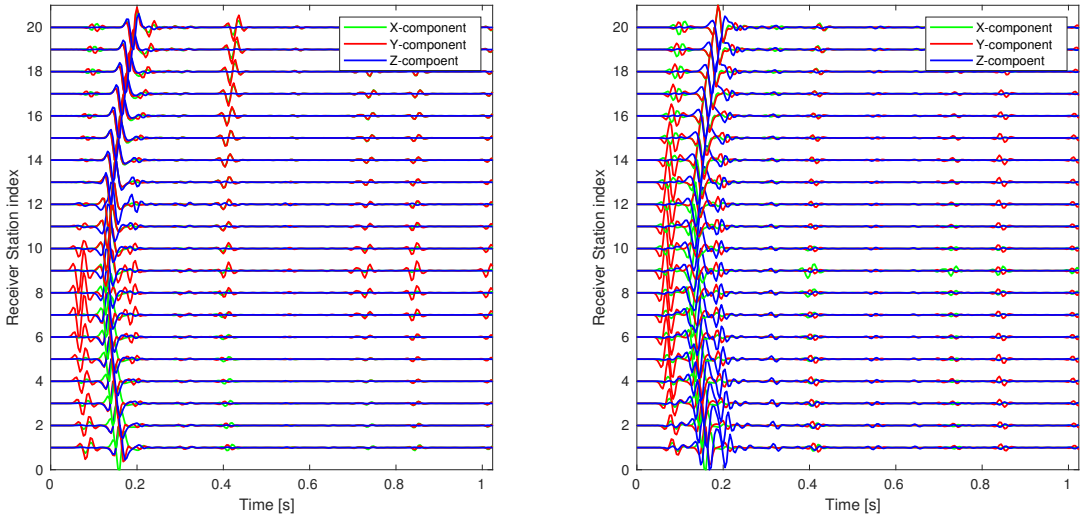


Figure 3-3. Synthetic traces using the SEAM Arid model for two different sets of angles: (a), $\phi = 56°$, $\delta = 15°$, $\lambda = -150°$; (b), $\phi = 12°$, $\delta = 78°$, $\lambda = 102°$.

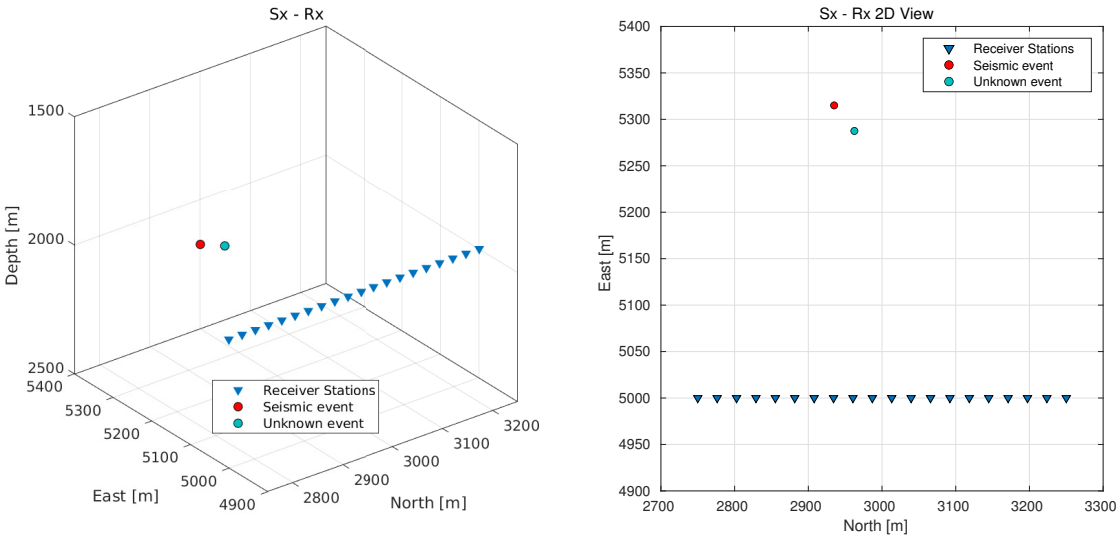
magnitude of such arrivals can change, and some components can have stronger amplitudes compared to the others. For example, the amplitudes of the P-wave arrivals seem to be stronger for the receiver stations # 10 to # 20 in Figure 3-3b compared to the P-wave arrivals in Figure 3-3a for the same set of receivers. Having events of the same magnitude with specific

fault orientations can yield seismograms with different amplitudes and consequently different moment tensors.

The seismograms shown in figures 3-3a, and 3-3b have a length of 1.024 s, corresponding to a total of 256 time samples N_s using a sampling rate of 4 ms. In the following sections, seismograms with 768 time samples (3.072 s) are considered as input to the ANN.

3-1-1 Training and prediction using one microseismic event

In this section, I explain the procedure for training an ANN to predict the moment-tensor components from synthetic microseismic data. First, let's consider the same acquisition geometry of Figure 3-1 and two microseismic events.



(a) **(b)**

Figure 3-4. 3D (a), and 2D (b) view of the acquisition geometry used for training one microseismic event.

The red point in Figure 3-4 represents the event corresponding to the seismograms that are used for the learning process. Subsequently, the trained network is used to predict the moment-tensor components from data generated by same event and also from data generated by a different one, represented in this case by the green point in Figure 3-4.

Earlier in this section it was described that synthetic seismograms can have different amplitudes depending on the orientation of the fault plane. As different fault orientations yield different moment tensors, it is desired to have a training dataset composed by seismograms containing the information from all the possible angle combinations (strike ϕ, dip δ, and rake λ) that can describe the geometry of a microseismic event. In the next example, a borehole microseismic dataset is generated using angles sampled every 5°, giving a total of 101,251 fault configurations for the red point.

Since for every combination of ϕ, δ, and λ, 20 three-component seismograms are obtained, (see figures 3-3a and 3-3b), the different fault orientations lead to a total of 6,075,060 seismograms.

For every set of angles the true moment tensor is calculated and stored for the training phase.

The generated seismograms are then preprocessed and used as inputs for the neural network. For this experiment, the data are split as follows:

- Training set: 40%.
- Validation set: 10%.
- Test set: 50%.

The training set is used to update the parameters of the network that minimizes the difference between the predicted and true values, and the validation set helps to keep track of the network's performance on non-analyzed data to avoid overfitting. Once the network is trained, the moment-tensor components are predicted on the test set. The results of the inversions can be depicted as scatter plots showing the difference between the predicted and true values, or by using the so-called "beach ball" diagrams (Shearer, 2009). Finally, the fault angles are also predicted from the resulting moment tensors. These results are shown in figures 3-5, 3-6, and 3-7.

Figure 3-5 shows the trained network fits well data generated by the same event achieving a prediction accuracy of 0.99. The bands displayed in Figure 3-6 for the three fault angles are due to the existence of two plane solutions: the actual fracture plane and an auxiliary plane that is obtained by swapping the normal and slip vectors. By adding additional constraints, e.g., previous geological knowledge, this ambiguity can be solved. The predicted angles that are shown in the plot correspond to the solutions giving the lowest error compared to true values. Lastly, the beach ball diagrams in Figure 3-7 show the true and predicted moment tensors of five random seismic events. The shaded red areas in the beach balls represent compressional motion at the location of the event, and the white areas account for the dilatational motion (Stein and Wysession, 2005).

As previously mentioned, the inversion results shown in figures 3-5, 3-6, and 3-7 were carried out using data generated by the same event. However, it is desired to predict the moment-tensor components using data from an event at a different position. The trained network from the previous example is now used to estimate the moment-tensor components of the event marked in green as shown in Figure 3-4. The results of the inversion are depicted in figures 3-8, 3-9 and 3-10.

As it can be seen in the inversion results, the neural network is not able to predict the moment-tensor components nor the fault angles using the learned information for an event located at a different position. Thus, the ANN must be trained using data generated by several seismic events at different locations in the space to better generalize the predictive capabilities of the network. It is however necessary to find the optimal distance of the seismic sources in the geometry of Figure 3-1.

3-2 Analysis of the optimal distance of the seismic sources

To find the optimal distance between the seismic sources used to train the ANN, four micro-seismic events are uniformly distributed in a Cartesian space at the edges of a square. Their

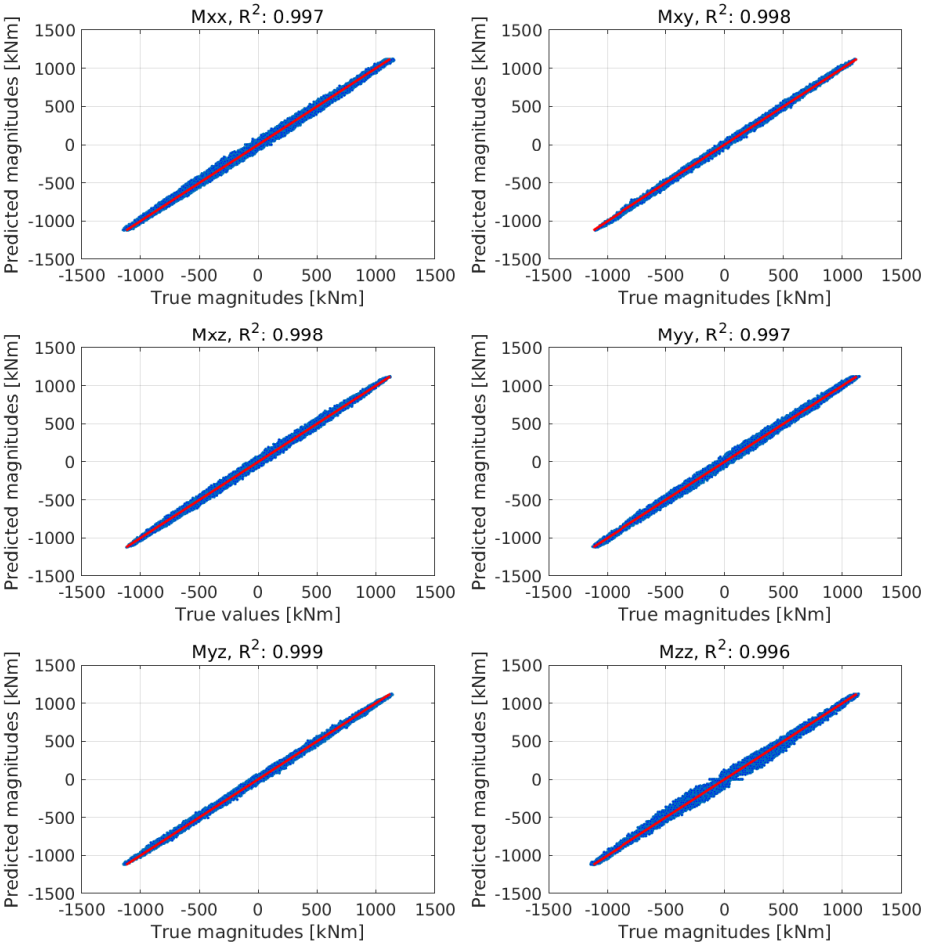


Figure 3-5. Predicted vs. true magnitudes of the different moment-tensor components; data generated by the same seismic event.

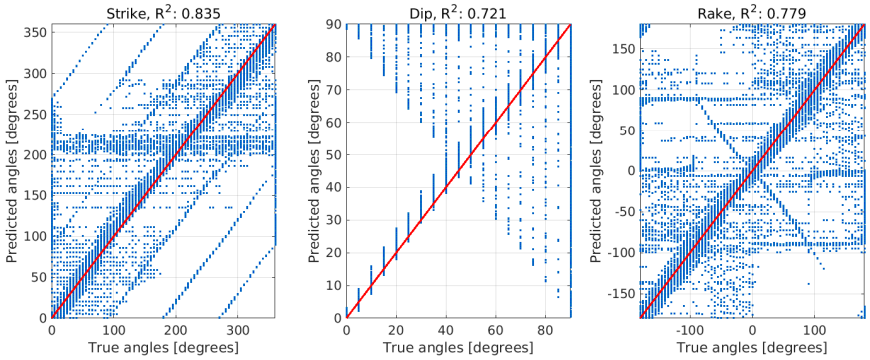


Figure 3-6. Predicted vs. true fault angles. Left: strike. Middle: dip. Right: rake; data generated by the same seismic event.

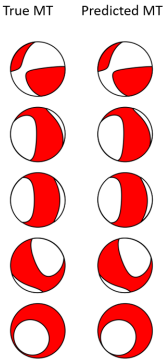


Figure 3-7. Beach ball plots for five random events: true (left) vs. predicted (right); data generated by the same seismic event.

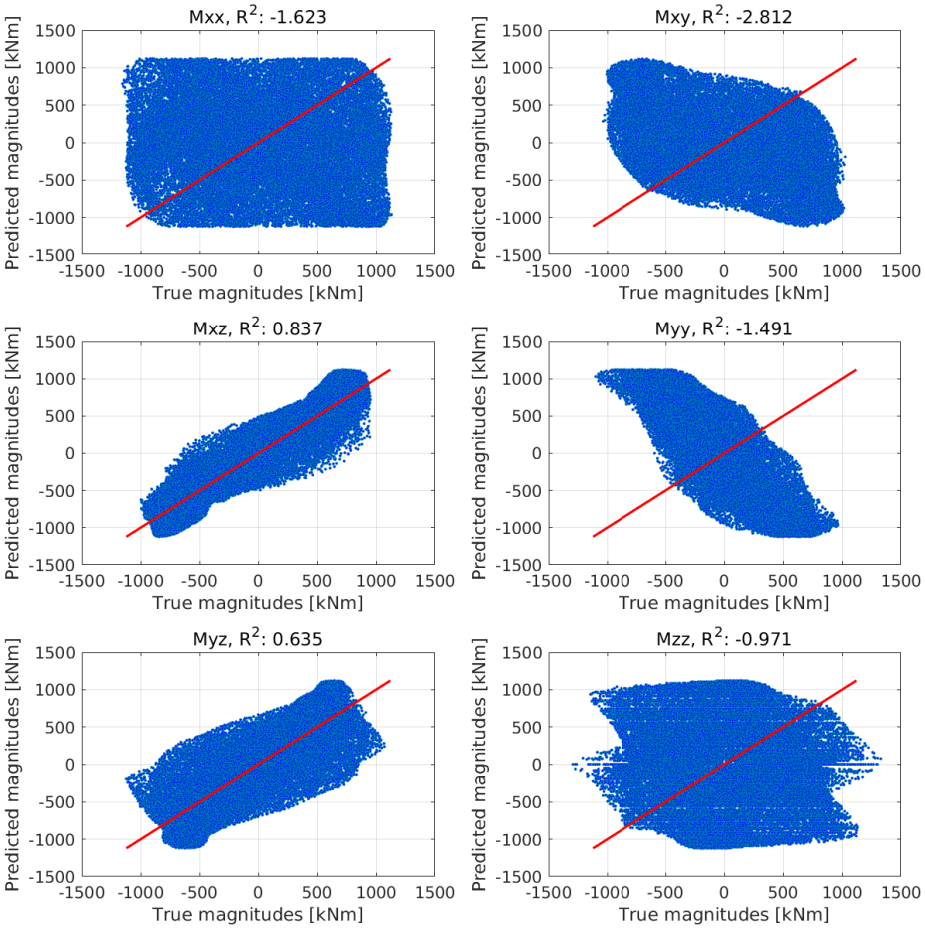


Figure 3-8. Predicted vs. true magnitudes of the different moment-tensor components; data generated by a different seismic event not used during the training.

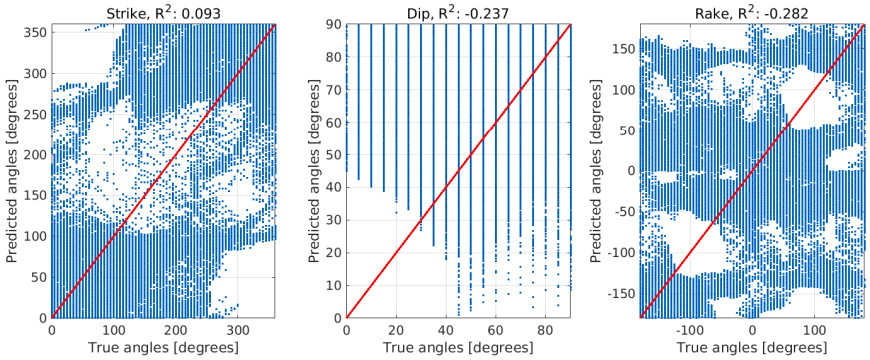


Figure 3-9. Predicted vs. true fault angles. Left: strike. Middle: dip. Right: rake; data generated by a different seismic event not used during the training.

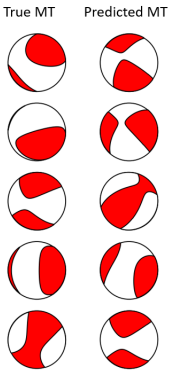


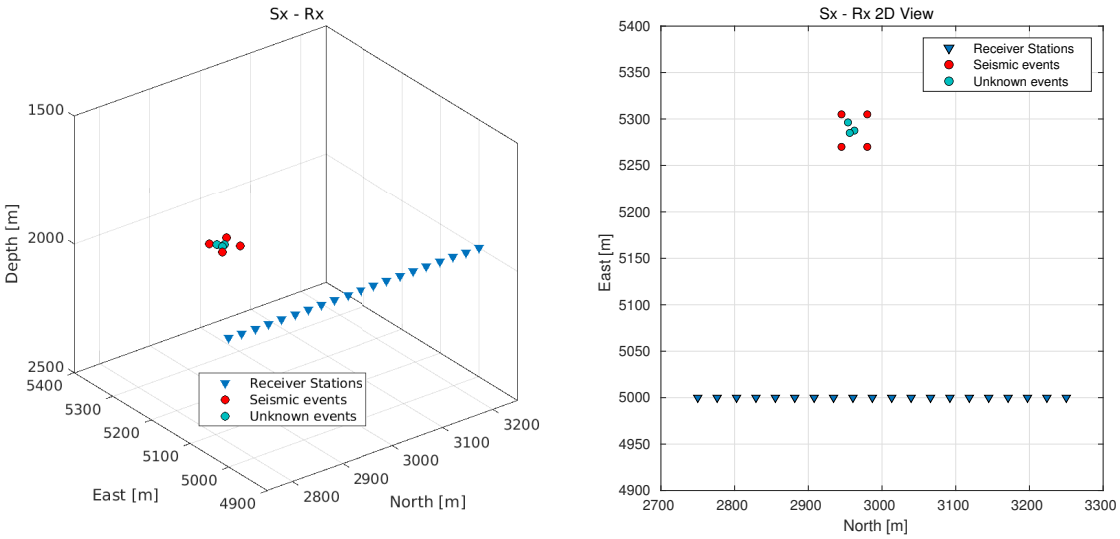
Figure 3-10. Beach ball plots for five random events: true (left) vs. predicted (right); data generated by a seismic event not used during the training.

distance from the receiver stations is about 300 m (see Figure 3-11a). The goal of this analysis is to predict the moment-tensor components and the fault angles of an event positioned inside such a distribution of sources. The neural network is trained using the seismograms generated by the four sources at the corners of the square, and then it is used for predicting the moment tensors of events positioned inside the array (green points in Figure 3-11b).

The problem now is to determine the optimal dimension of the square giving an acceptable prediction accuracy. Therefore, the training is carried out using four squares of different dimensions:

- Case A: $25\,m \times 25\,m$ square array.
- Case B: $30\,m \times 30\,m$ square array.
- Case C: $35\,m \times 35\,m$ square array.
- Case D: $40\,m \times 40\,m$ square array.

For each of these cases, the seismograms generated by the four sources at the corners of the square arrays are used to train the neural network. In Section 3-1-1, the network was trained



(a) **(b)**

Figure 3-11. 3D (a), and 2D (b) representations of the acquisition geometry and of the locations of the microseismic events.

using a dataset that contained seismograms from the 101,251 angles combinations. In this section, the generated waveforms for every source in the square have information generated by different sets of angles so that the network can recognize many fault orientations.

Furthermore, the resulting dataset needs to be split for the training into:

- Training set: 90%.
- Validation set: 10%.

The reason for these values is that the trained network is used to predict the moment-tensor components from data generated by an event at a different position. Hence, a test set is no longer necessary during the training phase as the information from the same events is not used for predicting the moment-tensor components.

After training the network, the moment-tensor components as well as the fault angles are predicted for an unknown event positioned at the center of the square. In figures 3-12, 3-13, and 3-14 the prediction results are displayed for the source distance of 25 m using a total of 6,000 random events.

As it can be observed, the neural network is capable of predicting the moment-tensor components of a different microseismic event not seen during the training phase. The prediction accuracy for each one of the components slightly decreases compared to the case where data from the same dataset was used to perform the prediction.

The same analysis is carried out for the three remaining cases. Figure 3-15 summarizes the results of the average prediction accuracy of the six moment-tensor components while increasing the dimension of the square array for cases A, B, C, and D. The prediction is performed

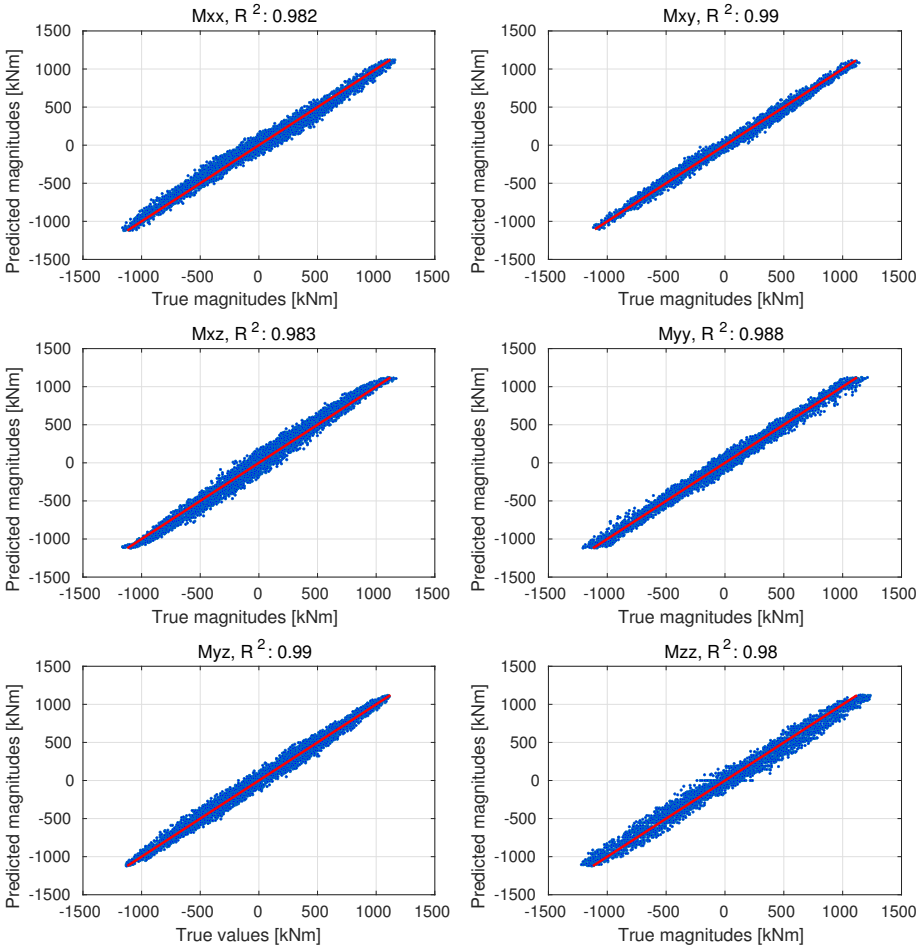


Figure 3-12. Predicted vs. true magnitudes of the different moment-tensor components; case A (distance = 25 m).

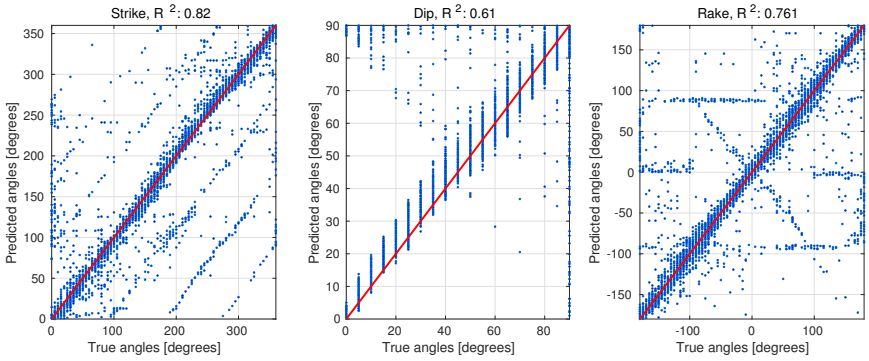


Figure 3-13. Predicted vs. true fault angles. Left: strike. Middle: dip. Right: rake; case A (distance = 25 m).

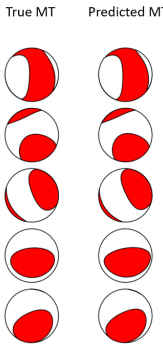


Figure 3-14. Beach ball plots for five random events: true (left) vs. predicted (right); case A (distance = 25 m).

using the same event at the same location but varying the distance of the surrounding sources with respect to the event in the center of the square.

The blue points marked in Figure 3-15 represent the average accuracies for the four studied cases. From these results, it can be seen that by increasing the distance from the known events with respect to the unknown one, the prediction accuracy drastically decreases. The neural network can yield accurate results for cases A and B (25 m and 30 m) with accuracies higher than 0.9. However, such value drops below 0.6 for cases C and D (35 m and 40 m).

From the data trend in Figure 3-15, I assume that the relation between the prediction accuracy and the dimensions of the square is linear. Thus, using the obtained prediction accuracies for the four dimensions of the square, a linear regression analysis can be performed to derive a relation between these two quantities:

$$\hat{a}_i = q + \Delta d_i, \tag{3-2}$$

where $\hat{a}_i$ are the estimated prediction accuracy values, d_i are the associated square dimensions, q and Δ are the line parameters to be calculated through regression, using a least-squares method, in this case. The results are represented by the red curve in Figure 3-15. From the calculated parameters Δ and q, an approximation of the prediction accuracy for the moment-tensor components of an unknown event can be obtained when using a uniform distribution of sources separated by a determined distance. In the analyzed example, satisfying results can be achieved for sources 25 m and 30 m apart from each other.

3-3 Analysis of the effect of the noise

In the analysis carried out in Section 3-2, the seismograms had infinite signal-to-noise ratio (SNR), meaning that they were not contaminated with possible noise sources. In real borehole seismic acquisitions, the recorded data is characterized by low SNR values due to the weak energy produced during the fracturing operations. In this section, the moment-tensor components are predicted using the uniform source distributions of 25 m and 30 m with the difference that white Gaussian noise is added to the seismograms.

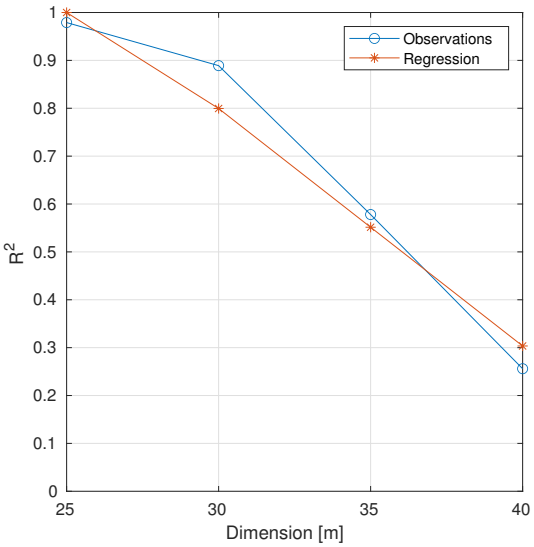


Figure 3-15. Prediction accuracy vs. dimension of the square array.

The SNR of a signal can be quantified as follows:

$$\text{SNR} = 10\log_{10}\left(\frac{E[s^2]}{E[n^2]}\right), \tag{3-3}$$

where the units of the SNR are given in decibels (dB), $E[s^2]$ is the expected value of a signal s, and $E[n^2]$ is the expected value of the noise n. The latter is the unknown quantity, and $E[s^2]$ can be calculated as follows:

$$E[s^2] = \frac{1}{N_s}\sum_{n=1}^{N_s}|s[n]|^2 \tag{3-4}$$

where N_s is the number of time samples of the signal. As the noise is considered to be normally distributed with zero mean, its expected value can be approximated to be equal to its variance, thus yielding the following relation between the SNR and the expected values of the signal and the noise:

$$SNR = 10\log_{10}\left(\frac{E[s^2]}{\sigma_n^2}\right), \tag{3-5}$$

where σ_n is the standard deviation of the Gaussian noise. Furthermore, equation (3-3) can be rewritten to calculate σ_n from the expected value of the signal and knowing the value of the SNR:

$$\sigma_n = \sqrt{\frac{E[s^2]}{(10^{(SNR/10)})}} \tag{3-6}$$

Given different SNR values, noisy seismograms are generated and used to train the ANN. This analysis is carried out for sources uniformly distributed at 25 m and 30 m (cases A and B) as those distances were the ones giving the best results from the distance analysis in Section

3-2. The inversion results for case A and an SNR of 0 dB (signal being of equal power as the noise) are shown in figures 3-16, 3-17, and 3-18.

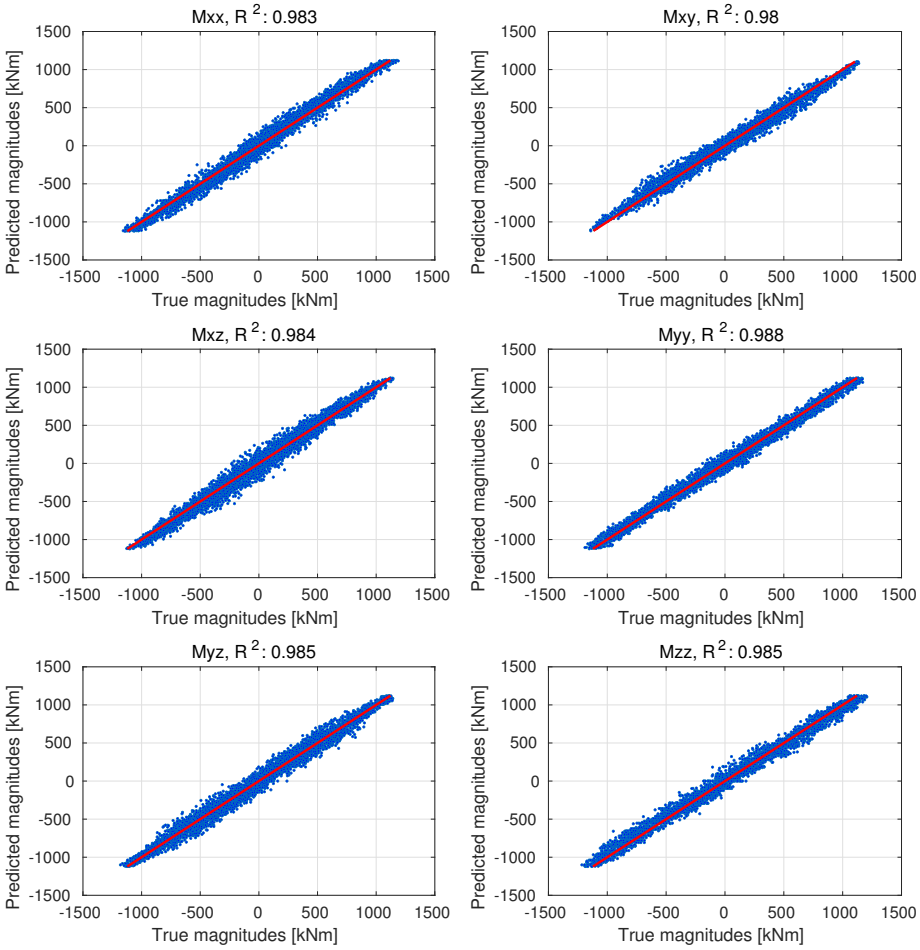


Figure 3-16. Predicted vs true magnitudes of the different moment-tensor components; case A (distance = 25 m) and 0 dB signal-to-noise ratio.

The predicted moment-tensor components in Figure 3-16 present accuracies close to 1 being 0.98 the lowest for the M_{xy} component. The strike and rake angles can still be predicted with accuracy values higher than 0.75. However, the accuracy is lower than 0.7 for the dip. These results show that the neural network is capable of giving acceptable results to predict the moment-tensor components when using seismograms that are contaminated with noise.

The same analysis is performed for different SNR values to analyze the behavior of the network when noisy traces are utilized. Gaussian noise is added to the traces for SNR values ranging from -20 dB to 20 dB at steps of 5 dB. The results of the average prediction for the six moment-tensor components compared to the different SNR values for the sources' distribution of 25 m and 30 m (case A and B) are shown in Figure 3-19. The blue curves show that the prediction accuracy decreases faster for negative values of the SNR, yielding more and more inaccurate

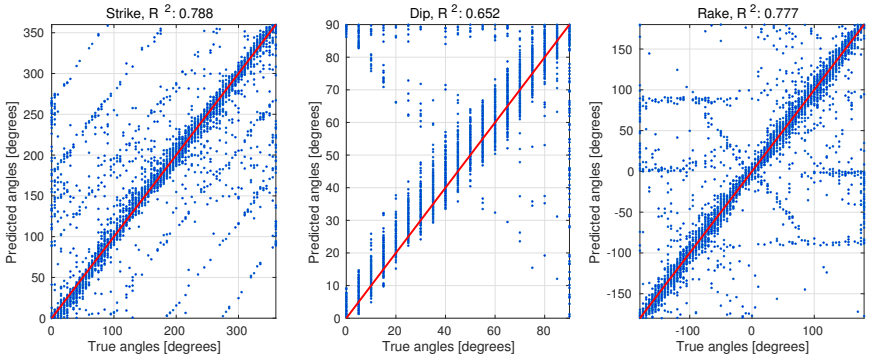


Figure 3-17. Predicted vs. true fault angles. Left: strike. Middle: dip. Right: rake; case A (distance = 25 m) and 0 dB signal-to-noise ratio.

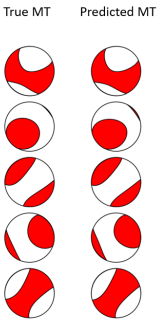


Figure 3-18. Beach ball plots for five random events: true (left) vs. predicted (right); case A (distance = 25 m) and 0 dB signal-to-noise ratio.

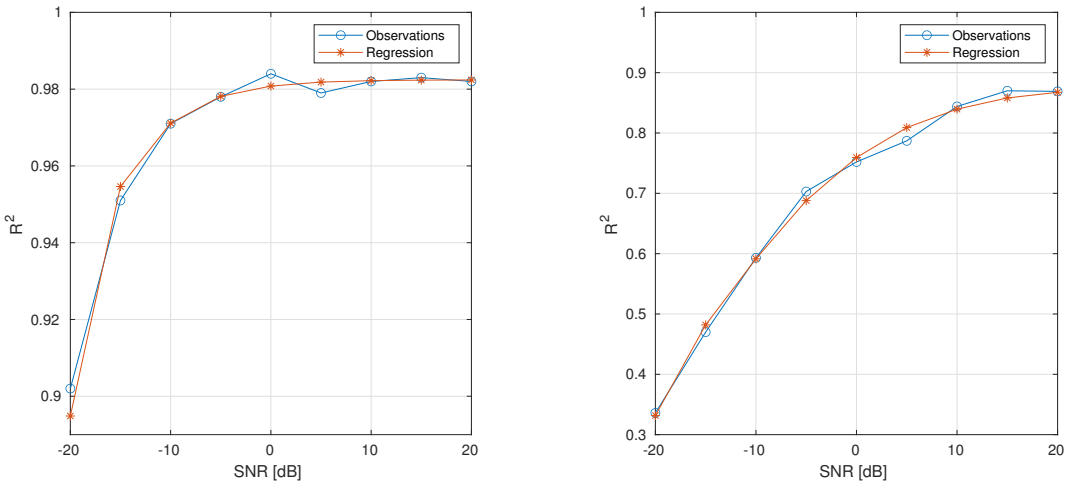


Figure 3-19. Regression results: accuracy vs. SNR: (a), case A (distance = 25m); (b), case B (distance = 30 m).

results of the estimated moment-tensor components. Moreover, the results for case B are not as good as for case A. The prediction accuracy for an SNR of 10 dB drops below 0.9 and for negative values of the SNR, to 0.8, whereas for case A it is above 0.9 for the considered range of the SNR.

Similar to the analysis on the relation between dimension of the square array and the average prediction accuracy in Section 3-2, an empirical relation between the SNR values and the accuracy can also be derived. As it can be seen from the blue curves in Figure 3-19a, the relation between the SNR and the prediction accuracy is non-linear. A function that can approximate the behavior of the data is the logistic function whose values range from 0 to 1. For example, for large values of the SNR, the prediction accuracy does not go beyond 1. Likewise, if the value of the SNR further decreases, it is expected to have inaccurate prediction results on the different moment-tensor components.

The logistic function can be defined as follows:

$$\sigma_n(x) = \frac{L}{1 + \exp(-k(x - x_0))}, \tag{3-7}$$

where the L, k, and x_0 are the parameters that need to be estimated. A non-linear least squares regression is performed to fit the logistic function to the observed data. From this analysis, the prediction accuracy for a specific source distribution can be estimated given any SNR value ranging from -20 to 20 dB. The results of the regression are illustrated by the red curve in Figure 3-19.

From the obtained results it can be appreciated that the observations fit well to the regression curve, meaning that an estimate of the prediction accuracy on the moment-tensor components can be obtained for the source distribution given in case A as shown in Figure 3-19a. Although the regression results for case B are accurate, for negative values of SNR the moment-tensor components cannot be accurately predicted. Therefore, the distance of the sources of 25 m is used for the subsequent analyses as it is the one giving the best prediction accuracy when noisy seismograms are used.

3-4 Analysis of complex source mechanisms

The distance and noise analysis in sections 3-2 and 3-3 were carried out considering pure double-couple (DC) events associated with shear faulting, which means that the microseismic event does not produce normal motion with respect to the fault plane. For a pure DC event, the first term on the right-hand side of equation (2-13) vanishes as the angle between the slip and normal vectors is 0°, yielding a moment tensor that has no contribution from the CLVD and ISO parts i.e., no volume changes are produced. In this section, the moment-tensor components given by equation (2-15) are used to obtain source mechanisms having shear and tensile motions due to a general dislocation source. To achieve that, the value of D_n must set to be non-zero. As an example, let's consider the characteristics of a microseismic event as detailed in Table 3-1.

By decomposing the moment tensor into its DC, ISO, and CLVD parts using equation (2-15) through eigenvalue analysis, the ISO and CLVD percentages are non-zero. These results are summarized in Table 3-2.

Table 3-1. Parameters of a shear-tensile microseismic event.

Parameter	Value
Moment magnitude (M_w)	-2
Rupture length (R)	1 m
Shear displacement (D_s)	0.131 mm
Normal displacement (D_n)	0.0131 mm
Lamé parameters	$\lambda = 14.24$ GPa $\mu = 7.509$ GPa
Fault angles	$\phi = 12°$ $\delta = 78°$ $\lambda = 102°$

Table 3-2. Moment-tensor parts of a shear-tensile event.

Component	Percentage %
Double-couple (DC)	69.903
Isotropic (ISO)	19.79
Compensated-linear vector-dipole (CLVD)	10.29

These results show that non-zero values for the normal displacements can yield moment tensors with non-DC components. It may look however that the value for the normal displacement in this example is not very significant, but a small change in it can have considerable changes in the moment-tensor components. In Table 3-1, the parameters of a microseismic event have been defined for a normal displacement of 0.0135 mm. If for example a microseismic event with normal displacement equal to 0.0263 mm is considered, the percentages of the different parts of the moment tensor can have significant variations. The results for such an event are shown in Table 3-3.

Table 3-3. Moment-tensor parts of a second shear-tensile event.

Component	Percentage %
Double-couple (DC)	51.26
Isotropic (ISO)	32.05
Compensated-linear vector-dipole (CLVD)	16.67

From the results in Table 3-3, it can be seen that incrementing the normal displacement from 0.0131 mm to 0.0263 mm, the percentage of the DC component of the moment tensor decreases approximately 20% compared to results shown in Table 3-1. Consequently, the percentages of the ISO and CLVD parts also increase.

Having defined seismic sources that can have non-DC contributions, the following test consists of determining the moment-tensor components of unknown events characterized by shear and tensile fracturing following the parameters of Table 3-4.

Table 3-4. Event parameters.

Parameter	Values
Moment magnitude (M_w)	-2
Rupture length (R)	1 m
Shear displacement (D_s)	0.131 mm
Normal displacement (D_n)	0.01-0.0131 mm

This analysis is carried out using the optimal distance of sources found in Section 3-2 to train four events distributed in a square, and noise-contaminated seismograms with an SNR of 0 dB. The trained network is subsequently used to predict the moment-tensor components and fault angles of an event positioned at the center of the square. The results of the inversion are shown in figures 3-20, 3-21, and 3-22.

From the obtained results it can be visualized that the magnitude of some moment-tensor components has increased compared to the results in sections 3-2 and 3-3. These changes can be better appreciated in the M_{xx}, M_{yy} and M_{zz} components as they are the ones associated with the ISO part of the moment tensor as described in equation (2-29). Further, from this test it has been proven that the neural network is capable of giving accurate results for moment tensors having non-DC components as the cloud of points shown for all the components slightly deviate from the true values given by the red line. Moreover, the beach balls diagrams depicted in Figure 3-22 also show accurate results when predicting the moment-tensor for five random events as small deviations are produced from the true beach balls.

Nevertheless, having non-DC components in the moment tensor considerably affects the prediction of the fault angles due to the deviation of the slip vector from the fault plane. Thus, for calculating the respective fault angles, it is necessary to take into account the effect of the angle α in the equations defining the normal and slip vectors (see equation (2-3)). This angle can be directly estimated from the eigenvalues of the predicted moment tensor using the following relation (Vavryčuk, 2011):

$$\sin\alpha = \frac{m_1 + m_3 - 2m_2}{m_1 + m_3}, \tag{3-8}$$

where m_1, m_2 and m_3 are the eigenvalues of the predicted moment tensor. The strike and dip can be directly estimated from the components of the normal vector. These angles and the estimated value of α are used to compute the rake from the equations defining the slip vector. The results presented in Figure 3-21 show a decrease in the prediction accuracy compared to the angles obtained in Section 3-3.

3-5 Storage optimization: inversion in the frequency domain

The space occupied on disk by the generated synthetic datasets is a crucial aspect to consider. A dataset containing waveforms from 101,251 different fault orientations has an approximate size of 37 GB. As the modelled seismograms in the previous sections used the same number of angle combinations, the resulting datasets were considerably extensive. Thus, finding alternatives to optimize the space required on the disk to store the information is essential.

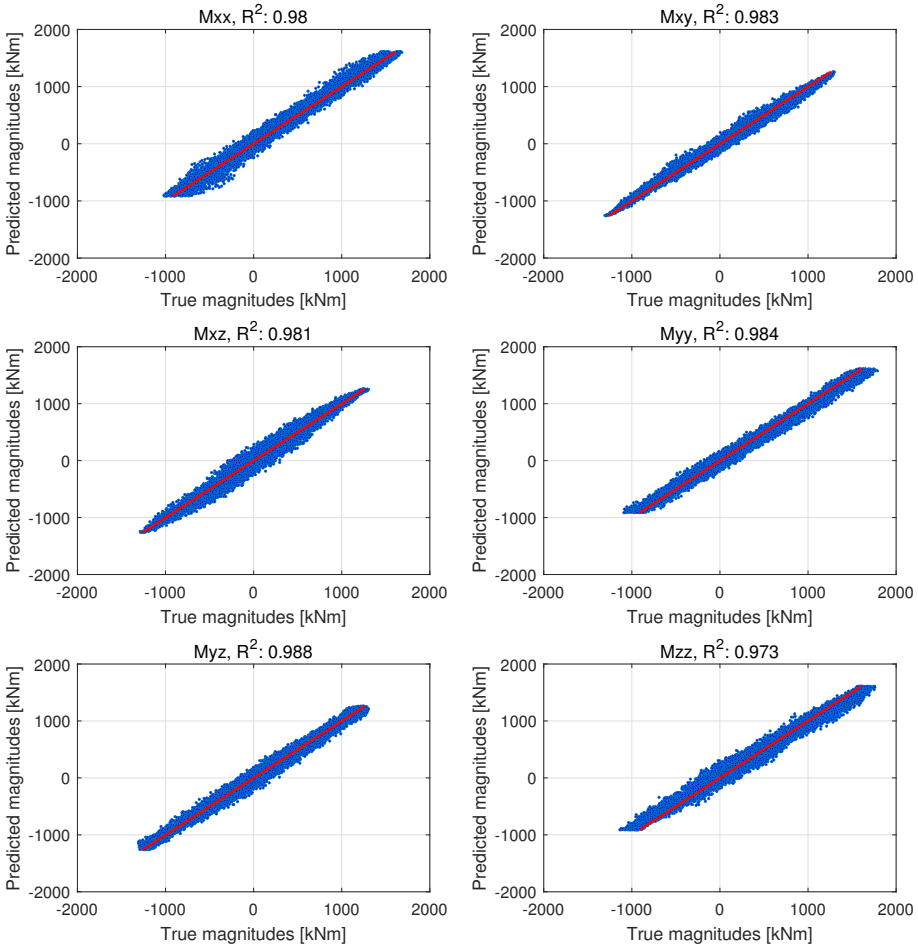


Figure 3-20. Predicted vs. true magnitudes of the different moment-tensor components; non-DC test.

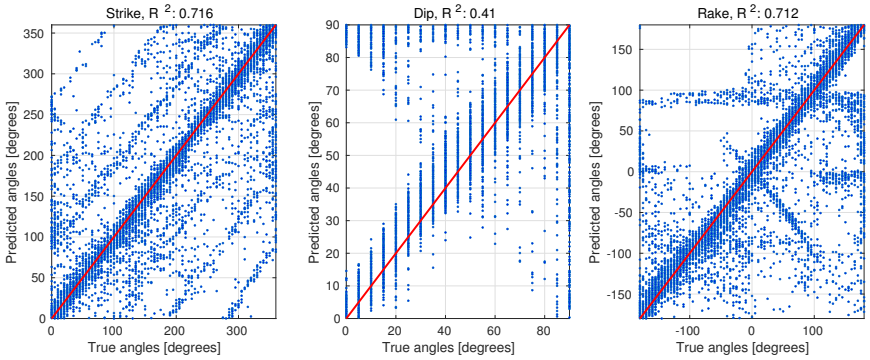


Figure 3-21. Predicted vs. true fault angles. Left: strike. Middle: dip. Left: rake; non-DC test.

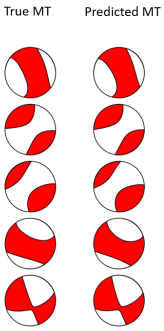


Figure 3-22. Beach ball plots for five random events: true (left) vs. predicted (right); non-DC test.

In this section, the inversion of the seismic moment tensor is performed in the frequency domain using the Fourier transform of the observations. Figure 3-23 shows an example of the vertical component of a seismogram generated by a random microseismic event and its amplitude spectrum.

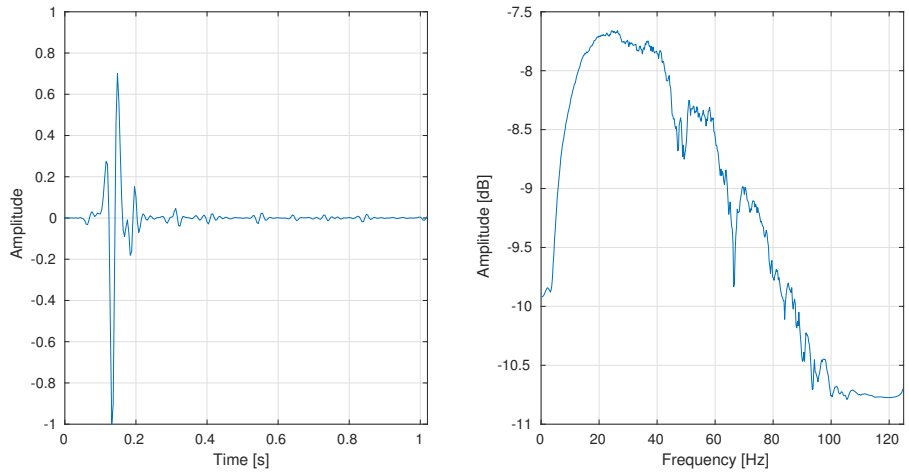


Figure 3-23. Vertical component seismogram (left), and its amplitude spectrum in dB (right).

The full bandwidth of the signal ranges from 0 Hz to 125 Hz. It can be appreciated in the plot of the amplitude spectrum that a considerable part of the signal lies within a narrower band. Thus, part of the frequency components of the seismograms can be filtered and subsequently fed to the neural network to perform the inversion of the moment tensor. The information that is used as input to the network is the real and imaginary parts of the Fourier coefficients of the seismograms limited in frequency.

In this thesis, two inversions in the frequency domain are performed using two different bandwidths: 5 Hz - 80 Hz and 15 Hz - 70 Hz. These two bandwidths are better visualized in Figure 3-24. The red and green lines in both plots show the low and high-cut values of the two chosen bandwidths. The inversion is then performed with the real and imaginary parts of the Fourier coefficients of the seismograms using the selected frequencies for each case.

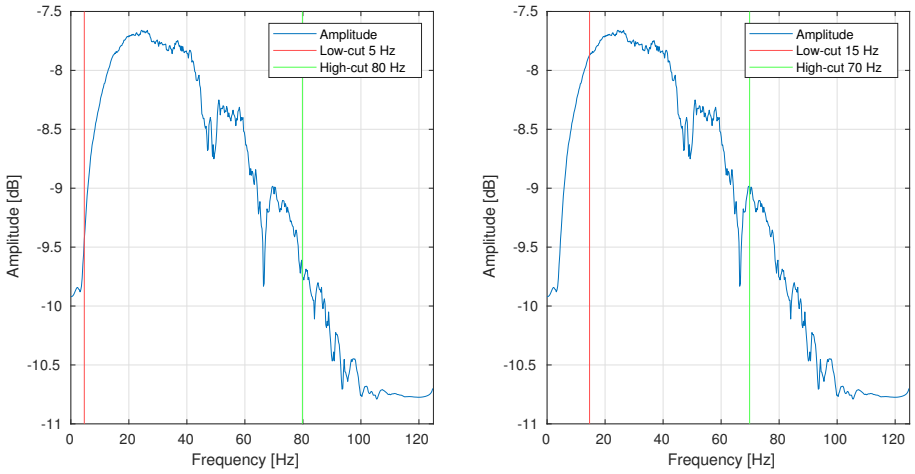


Figure 3-24. Bandwidth 5 Hz - 80 Hz (left); bandwidth 15 Hz - 70 Hz (right).

In Section 3-4, the neural network was trained using four sources uniformly distributed with a separation distance of 25 m between sources. In this section, the same distribution of sources is used with seismograms contaminated with Gaussian noise with an SNR of 0 dB, and moment tensors with non-DC components using the same parameters described in Table 3-4. The Fourier transform is calculated for all the seismograms and the bandwidths shown in Figure 3-24 have been used for the training. Similarly, the Fourier coefficients of the seismograms from the same bandwidths and generated for an unknown event inside the square have been used to predict its moment-tensor components. The results of the inversion for the two chosen bandwidths are depicted in figures 3-25, 3-26, 3-27, 3-28, 3-29, and 3-30.

From the results it can be observed that all the moment-tensor components have a prediction accuracy higher than 0.9. For some components in Figure 3-28, the accuracy slightly decreases as less Fourier coefficients are used during the inversions. For instance, by comparing the results for the component M_{zz} with the ones obtained in Section 3-4, the prediction accuracy drops from 0.973 to 0.938. This means that the moment-tensor components can be still predicted with acceptable results using the trained network.

The main purpose of carrying out the inversion in the frequency domain is to reduce the size of the generated dataset. Using the full bandwidth of the signal during the inversion, yields a dataset of 37 GB. When using Fourier coefficients in a narrow bandwidth, the size of the dataset can be reduced to 22 GB if a bandwidth of 5 Hz - 80 Hz is used and to 16.5 GB for a bandwidth of 15 Hz - 70 Hz. The latter represents a reduction of more than half of the original size and can still yield accurate results.

3-6 Inversion results on a realistic scenario

The results obtained in the sections 3-2, 3-3, and 3-4 showed that the seismic moment tensor with non-DC components can be predicted using noisy seismograms extracted from a microseismic event not used during the training phase. The obtained information from the previous analysis is used to train a set of 121 seismic sources having non-DC components and

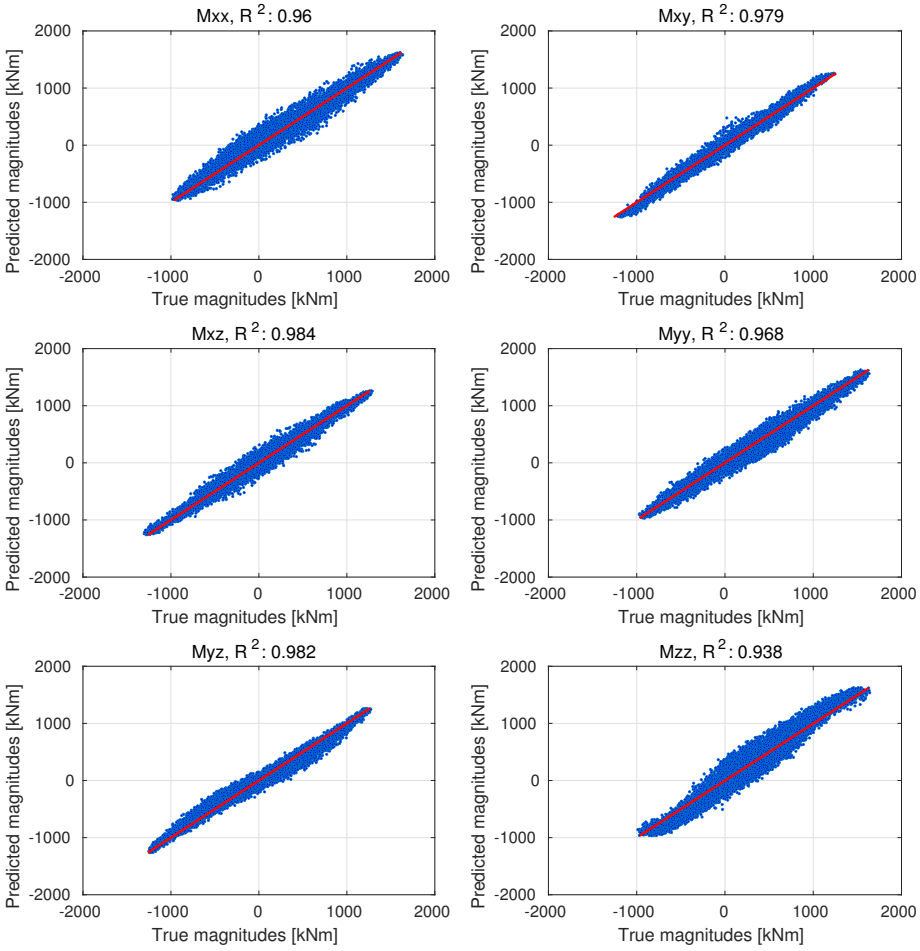


Figure 3-25. Predicted vs. true magnitudes of the different moment-tensor components; bandwidth: 5 Hz - 80 Hz.

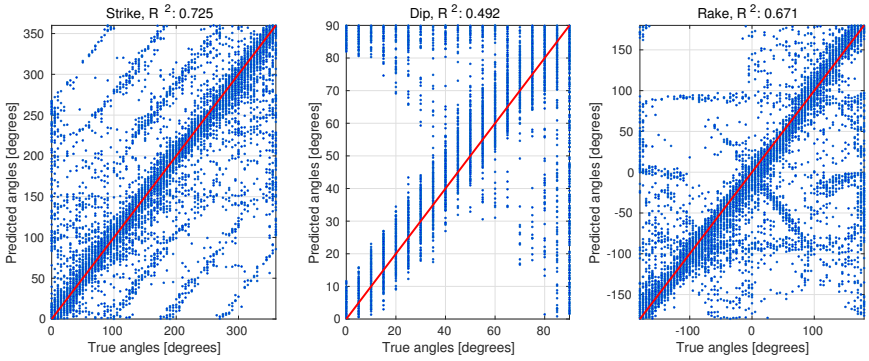


Figure 3-26. Predicted vs. true fault angles. Left: strike. Middle: dip. Left: rake; bandwidth: 5 Hz - 80 Hz.

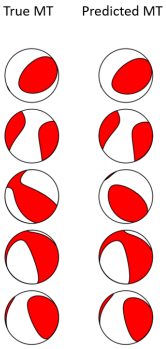


Figure 3-27. Beach ball plots for five random events: true (left) vs. predicted (right); bandwidth: 5 Hz - 80 Hz.

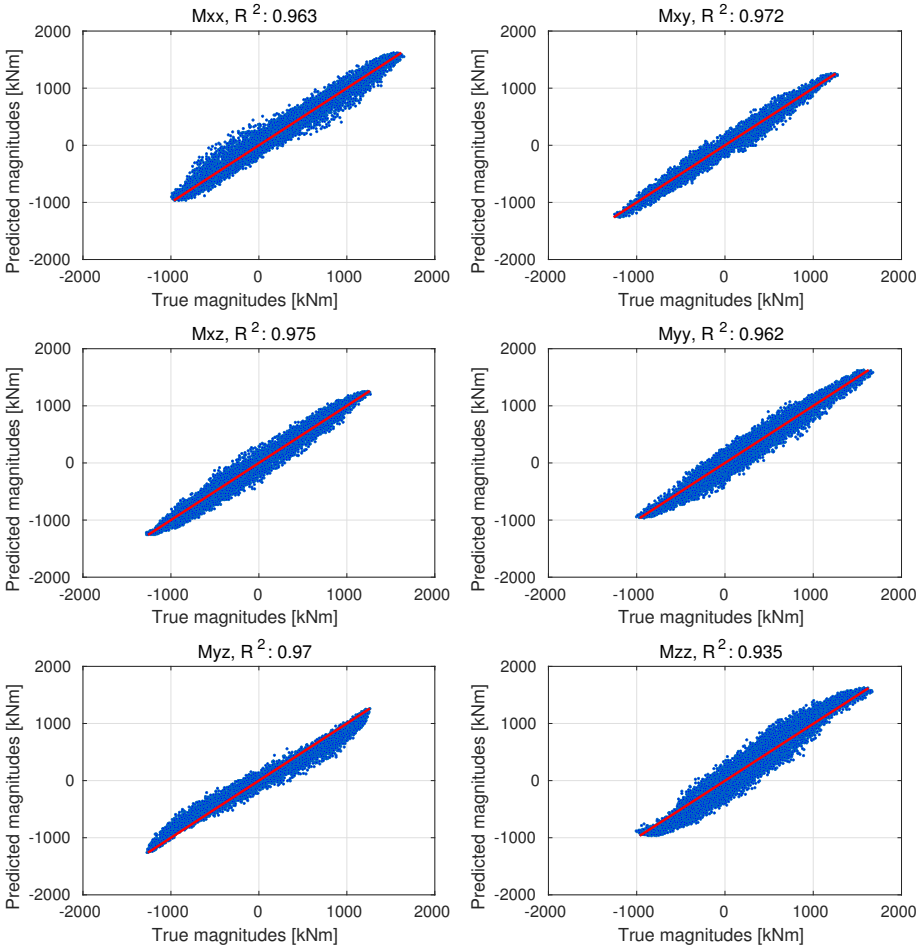


Figure 3-28. Predicted vs. true magnitudes of the different moment-tensor components; bandwidth: 5 Hz - 70 Hz.

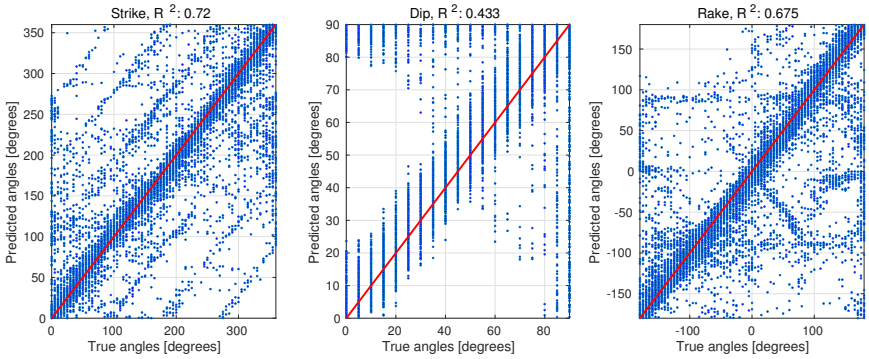


Figure 3-29. Predicted vs. true fault angles. Left: strike. Middle: dip. Left: rake; bandwidth: 15 Hz - 70 Hz.

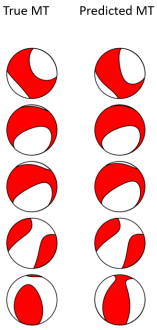


Figure 3-30. Beach ball plots for five random events: true (left) vs. predicted (right); bandwidth: 15 Hz - 70 Hz.

uniformly distributed using a distance of 25 m between the sources. Further, Gaussian noise is also added to the seismograms so that an SNR of 0 dB is obtained.

The sources are assumed to be positioned 250-500 m aside of the receiver array with three additional unknown events collocated inside the area covered by the sources (Figure 3-31). The neural network is trained using seismograms from the 121 sources for the subsequent prediction of the moment-tensor components of the three unknown events.

When the neural network was trained using the uniform distribution described in Section 3-2, the angles were sampled in a such way that every source in the dataset contained information from different combinations of angles. In this section a similar approach is followed to sample the angles so that every source shown in Figure 3-31 has a different combination of the strike, dip and rake. Finally, the range of normal displacement values used to obtain non-DC contributions for the different moment tensors is the same as in sections 3-4 and 3-5.

To further improve the training of the neural network and prevent overfitting, two *dropout* layers are added to the network architecture defined in Section 2-2-1. Dropout is a well-known regularization method used to reduced overfitting during the training. The concept behind the dropout method is straightforward: for every epoch during the learning process, every neuron has a probability of being temporarily ignored or "dropped" (Hinton et al., 2012). This means that the seismograms are not forward/backpropagated through that neuron, but it can be active during the next epoch. The probability (P) of dropping out a neuron at each

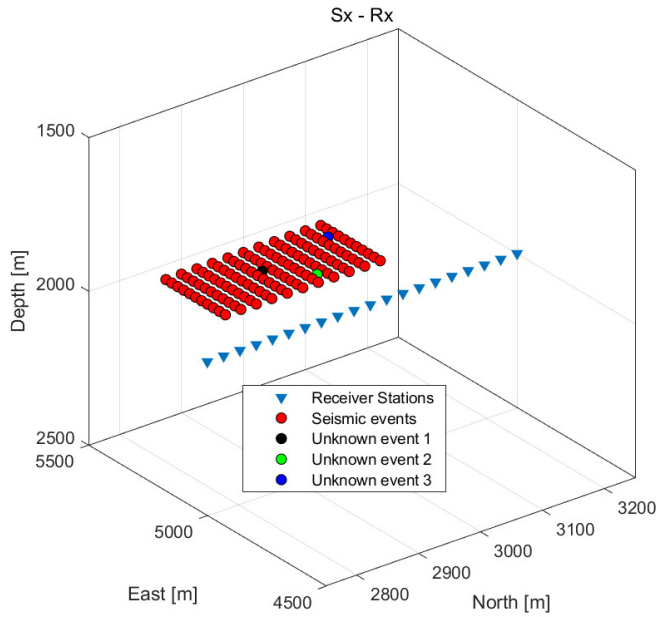


Figure 3-31. Geometry setup using a distribution of 121 microseismic events, and 3 unknown events.

epoch is a hyperparameter that needs to be set before training the network. The updated architecture is summarized in Table 3-5. In the previous sections the dropout layers were not used during the training due to the lower complexity of the problem.

Table 3-5. Artificial Neural Network architecture with dropout.

Layer	# Units	Activation function	# Parameters	Description
Input layer	$3 \times N_r \times N_s$	-	$3 \times N_r \times N_s$	$N_r = 20\,, N_s = 768$
1^{st} hidden layer	164	$\tanh(\cdot)$	$3 \times N_r \times N_s \times 164$	-
Dropout layer	-	-	0	$P = 15\ \%$
2^{nd} hidden layer	92	$\tanh(\cdot)$	164×92	-
Dropout layer	-	-	0	P= 15 %
3^{rd} hidden layer	64	$\tanh(\cdot)$	92×64	-
Output layer	6	-	64×6	-

The results for the predicted moment-tensor components and corresponding beach ball diagrams for the three unknown events are shown in figures 3-32, 3-33, 3-34, 3-35, 3-36, and 3-37.

The predicted moment-tensor components show a slight decrease in the prediction accuracy compared to the tests carried out in the previous sections. The prediction accuracy shown in figures 3-32, 3-34, and 3-36 for the components M_{xz} and M_{yz} are higher than 0.95, and for the remaining components is closer or higher than 0.8. Further, the beach ball diagrams illustrated in figures 3-33, 3-35, and 3-37 also show a slight deviation from the true moment

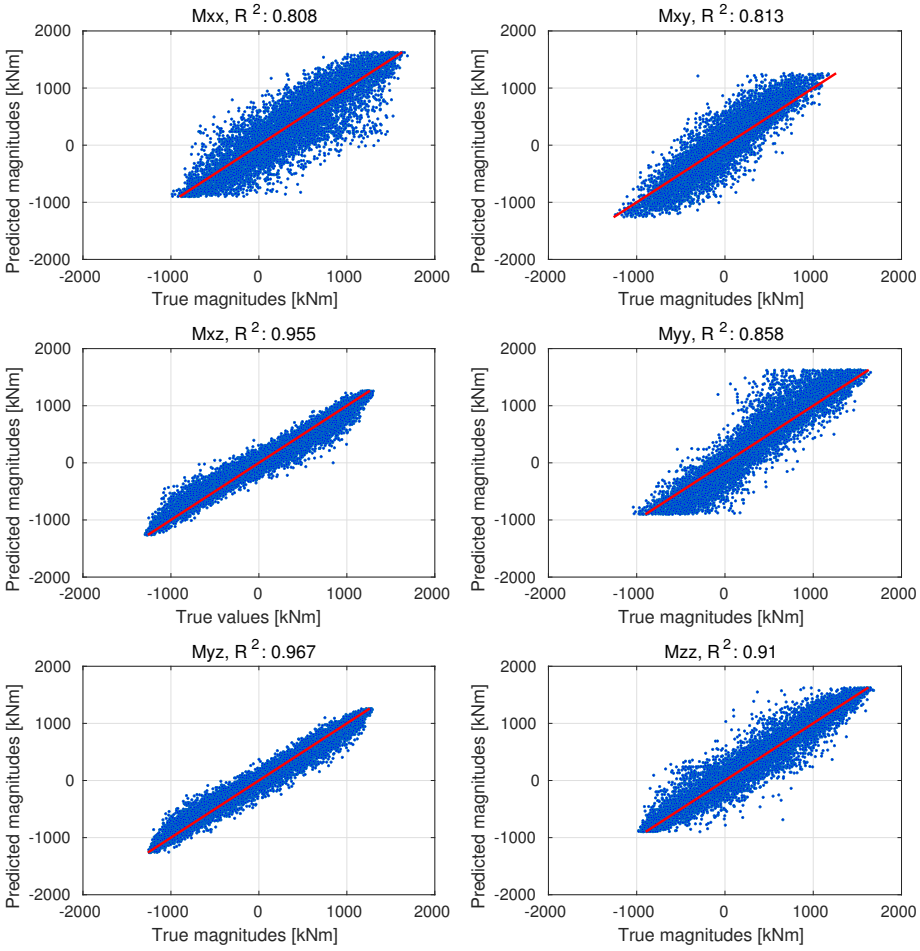


Figure 3-32. Predicted vs. true magnitudes of the different moment-tensor components; event 1.

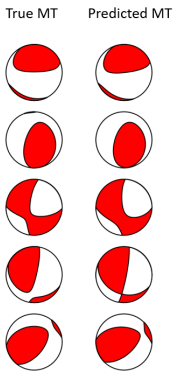


Figure 3-33. Beach ball plots for five random events: true (left) vs. predicted (right); event 1.

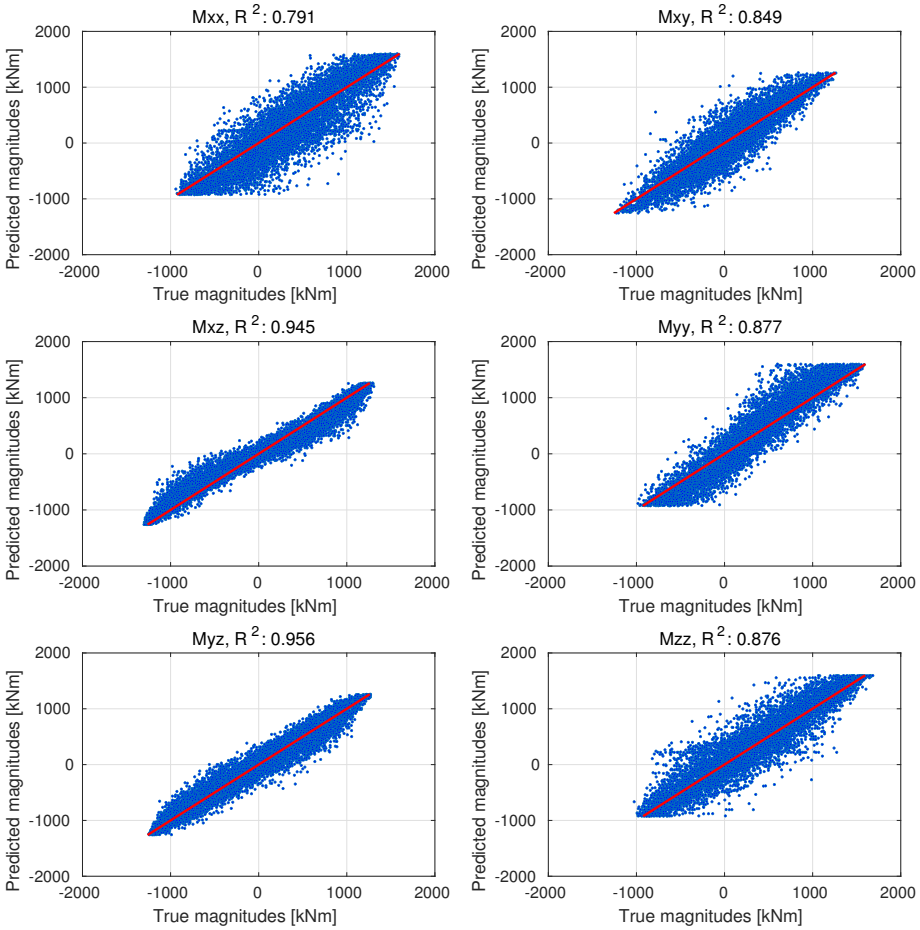


Figure 3-34. Predicted vs. true magnitudes of the different moment-tensor components; event 2.

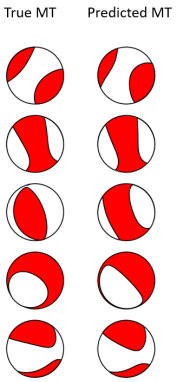


Figure 3-35. Beach ball plots for five random events: true (left) vs. predicted (right); event 2.

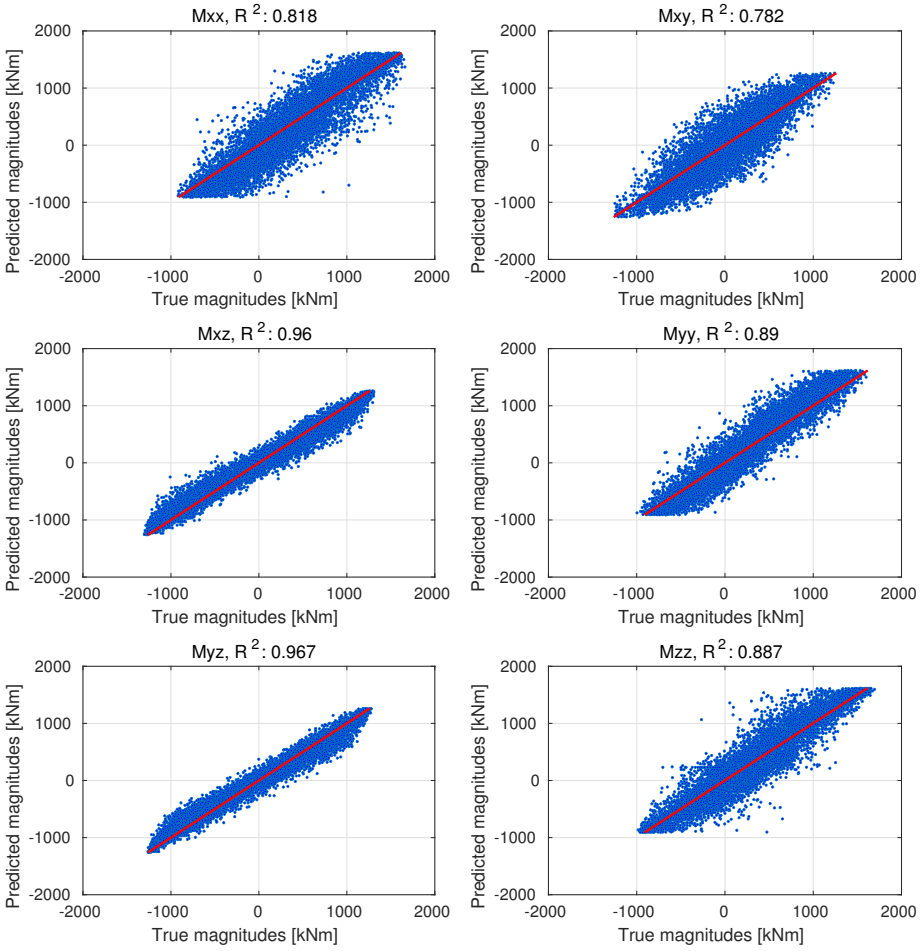


Figure 3-36. Predicted vs. true magnitudes of the different moment-tensor components; event 3.

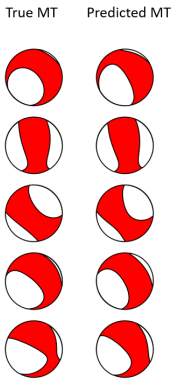


Figure 3-37. Beach ball plots for five random events: true (left) vs. predicted (right); event 3.

tensor. The decrease in the prediction accuracy may be associated with the high amount of information that the network needs to learn from the seismograms generated by the different seismic events.

The presence of noise in the seismograms is also a determinant factor to consider. Seismic signals with low SNR are prevalent in borehole acquisitions for fracking operations. Thus, before utilizing the recorded seismograms for further processing and imaging steps, it is crucial to apply denoising methods to boost the SNR of the signal (Iqbal et al., 2018). As an SNR of 0 dB was used to carry out the previous test, it is expected to have deviations from the predicted moment-tensor components as different noise realizations are added to each receiver station's seismograms. The neural network is however capable of yielding satisfactory results even when seismic traces are contaminated by noise.

Moreover, the non-DC components in the moment tensors also have an influence on the results. This can be noticed by the decrease of the prediction accuracy values in the M_{xx}, M_{yy}, and M_{zz} components of the three events associated with the ISO part of the moment tensor. As previously discussed in Section 3-4, small deviations of the normal displacement can have a significant change in the magnitude of the moment tensor components.

3-7 Computational considerations

The inversion results presented in this chapter were carried out using the following software and hardware specifications:

- Operating system: CentOS 7-9.
- Architecture: x86_64.
- CPU op-mode: 32-bit, 64-bit.
- CPU(S): 40.
- Processor model: Intel (R) Xeon (R) CPU E5-2690 v2 - 3.00 GHz.
- Thread(s) per core: 2.
- Core(s) per socket: 10.
- Socket(s): 2.
- Memory (RAM): 256 GB.

Generating a microseismic borehole dataset using the DWM can be computationally expensive. In the work carried out by Carrizo Mascarell (2020), the synthetic seismograms used for training the neural network were generated using a homogeneous velocity and density model. However, in this thesis, a 1D profile of 600 layers from the SEAM Arid model is used. Using such a complex model increases considerably the computational time of the forward modelling calculations.

Using the SEAM Arid model to calculate the Green's functions derivatives rises the computational cost of about 20 times when compared with the calculations using a homogeneous velocity and density models. For complex scenarios the code shows a significant increase in the dimension of different matrices accounting for the number of utilized sources, the number of receiver stations, the number of frequency points, and the total displacements calculated per receiver station. Such high dimensional matrices bring memory storage issues and an increase of computational times to perform the forward modelling calculations through vector-matrix or matrix-matrix operations.

The modelled Green's functions are subsequently convolved with known source moment tensors to yield the synthetic waveforms. Throughout the sections of this chapter, the neural network was trained considering a range of 101,251 combinations of angles to model different fault orientations. Using such a range of angles implies that the same number of convolutions needs to be performed to compute the seismograms. The computational time to generate three-component seismograms for 101,251 different combinations of (ϕ, δ, λ) is about 74 min on the machine previously described.

The ANN learning time can vary depending on the training dataset. In Section 3-1 the ANN was trained with the smallest dataset, including only the seismograms generated by a single microseismic event. In sections 3-2, 3-3, 3-4, and 3-5 the size of the training datasets increased of factor of 2 and the corresponding computational times also raised. The final test of Section 3-6, a complex example of training with seismograms generated by 121 microseismic events, showed the largest learning time.

Finally, the prediction times are similar for all the analyzed examples.

A summary of the computational times for training the neural network and predicting the moment-tensor components for the shown examples are summarized in Table 3-6. The average training time is about 12 min and the average prediction time is of about 18 s.

Table 3-6. Simulation times - inversion results.

Test	Training time	Prediction time
Analysis using a single source	6.803 min	25.03 s
Distance analysis	11.75 min	17.35 s
Noise analysis	10.612 min	17.21 s
Non-double-couple analysis	11.33 min	18.56 s
Frequency domain analysis	7.84 min	11.26 s
Analysis on a complete realistic scenario	23.48 min	20.45 s

Chapter 4

Conclusions and Discussion

The seismic moment tensor is a useful mathematical approximation for describing general seismic sources. It has become a helpful tool for monitoring microseismic events produced in energy-related environments such as hydraulic fracturing, geothermal energy extraction, CO_2 injection, deep-well waste isolation, among other applications. Surface and borehole seismic surveys are standard in microseismic monitoring, being the latter the preferred one for obtaining higher resolution results. The information obtained from borehole acquisition can be used to extract relevant properties of the target. Recorded data from borehole receivers can be inverted to estimate the seismic moment tensor, which gives a better understanding of the distribution of forces and rupture mechanisms at the location of the seismic event.

The estimation of the seismic moment tensor can be carried out in several manners. In this thesis, I focused on the inversion of full-waveform data using a deep feedforward neural network. The datasets used to train the neural network were generated employing the discrete-wavenumber method, where Green's functions are modelled in an elastic medium using a pre-defined velocity and density model. In this thesis, the SEAM Arid model was used to simulate microseismic signals produced from known moment tensors of low magnitudes, such as those encountered in hydraulic fracturing.

As downhole receiver arrays usually suffer from having inadequate angle coverage, resolving the six independent moment-tensor components using data recorded from a single well may not be possible due to the ill-posedness of the problem. However, this issue can be overcome by producing synthetic data containing information from many fault orientations. Subsequently, the data are fed to the neural network to estimate the seismic moment tensor.

To properly train the neural network, a sufficient amount of data needs to be used. Different moment tensors produce seismograms with different amplitudes due to the variation in the fault angles. Therefore, a dataset containing different sets of angles is essential. To achieve this I generated datasets from several moment tensors using angles sampled every 5°, yielding 101,251 possible combinations.

I have shown that the neural network can recognize the moment tensor from microseismic events located at different positions than the ones used for the training. In particular, I

proposed an array of four sources equally distributed at the edges of a square with variable distance from source to source. I then trained the network using the sources from the square array to predict the moment-tensor components and fault angles of an event positioned inside the square. This analysis was used to derive the larger source spatial sampling producing reasonable results, which was 25 m in this particular case. Moreover, I have shown that seismic traces with a signal-to-noise ratio of 0 dB can also be utilized to train the network and yield accurate results for the six components of the moment tensor and the respective fault angles. I derived an empirical rule to link the prediction accuracy with the data signal-to-noise ratio.

A big challenge in microseismic monitoring and earthquake seismology is the study of moment tensors with non-double-couple components. These are associated with tensile and compressive motions along the rupture plane where the event was produced. Artificial non-double-couple components can be obtained during moment-tensor inversion due to simplifications in the source model or poor station coverage. In this thesis, a moment-tensor model that accounts for tensile motion in a microseismic event was used to model seismograms due to sources with non-double-couple components. I used such seismograms to train the neural network and I successfully predicted the moment-tensor components due to sources with non-double-couple mechanisms located at different positions than those used for the training.

I proposed to carry out the inversion using the real and imaginary parts of the Fourier coefficients of the seismograms in a narrow band to optimize the space required on disk to store the datasets. I modelled seismic sources distributed every 25 m, a 0 dB signal-to-noise ratio, and non-double-couple mechanisms. I showed the inversion results in the frequency domain for two different bandwidths: 5 Hz - 80 Hz and 15 Hz - 70 Hz. The predicted moment-tensor components were obtained for microseismic events at different locations, showing prediction accuracies higher than 0.9 for both bandwidths.

An analysis of the computational times required in the different stages of the moment tensor prediction with ANN has also been carried out.

During hydraulic fracturing operations, multiple microseismic events can be triggered due to the pressure of the injected fluids. To emulate a similar scenario, I generated microseismic data from 121 different sources, each one having different fault angles, noise realization, and tensile motion to generate non-double-couple mechanisms. I used the resulting dataset to train the neural network and predict the moment-tensor components for three different events. I showed that the prediction accuracy for some of the components (M_{xx}, M_{zz}) dropped to 0.8, which may be due to the complexity of the seismic information used to train the neural network. Such complexity may come from variations on the location of the sources, different noise realizations on each receiver station, and the presence of non-double-couple components.

4-1 Additional original contributions

Besides the extension of the moment tensor inversion workflow for applications on realistic scenarios and the related analysis, I have also studied and implemented the following:

1. I have extended the forward modelling code that generated synthetic seismograms to account for source moment tensors with non-double-couple mechanisms, and slip vectors

that deviate from the fault plane. This deviation is quantified using the angle α as described in equation (2-3).

2. I have included in the code the formulas to calculate the scalar seismic moment, approximate rupture length of the fracture and its area given a specific moment magnitude described in equations (2-16), (2-17), and (2-18).

3. I have extended the forward modelling code to perform more calculations in parallel. Previous versions of the code utilized parallelization to calculate synthetic seismograms for different sets of angles (strike, dip, and rake). I have extended this approach so that different moment magnitudes and normal displacements can also be tested in parallel.

4. I have carried out an extensive investigation of the available hyperparameters of the implemented ANN. The parameters that I tested were the following: batch sizes, learning rates for the RMSprop optimizer, number of hidden neurons, number of hidden layers, dropout rates.

5. I have implemented additional functions to pre-process the synthetic seismograms before feeding them to the neural network. These functions include: adding Gaussian white noise to the seismograms to obtain a specific SNR value, data normalization, a more straightforward approach to split the data for training, validation and testing.

6. Implementation of the inversion workflow in the frequency domain to limit the bandwidth of the analyzed signals.

7. I have changed the data file format of the synthetic seismograms to HDF5. This format significantly improved the time needed to store and load the data for the respective inversions.

4-2 Future developments

The propagation medium considered in this thesis was isotropic and non-dispersive. However, microseismic monitoring activities are conducted over more complex media where some lithologies may exhibit anisotropic behavior. Thus, an extension to the present work could be related to the use of different anisotropic models, e.g., vertically transverse isotropic (VTI) media, which would require an upgrade of the current forward modelling code.

Deep feedforward neural networks have proven to be effective and robust to predict the seismic moment tensor using three-component seismograms even in the presence of Gaussian noise. Other alternatives to carry out the inversion could involve different network architectures such as convolutional neural networks. These might be used to estimate the seismic moment tensor and other variables of interest, such as the moment magnitude or the location of the event.

As the moment tensor inversion has been carried out using constant moment magnitudes for simulated microseismic events, a potential extension is to test the inversion using events with a broader range of moment magnitudes as the values encountered in microseismic monitoring can vary significantly. For predicting moment tensors of different magnitudes, an extensive amount of data may be needed for the neural network to learn different ranges of magnitudes.

Another possible future development of this work is to test the trained neural network on field data. This would require a thorough analysis of the seismograms in terms of phase and amplitudes, signal-to-noise ratio and frequency content, when compared to synthetic data. Estimating the seismic moment tensor from field data can significantly impact real-time microseismic monitoring in the development of unconventional reservoirs. It can help to understand better the stress distributions and rupture mechanisms occurring during fracking operations.

Bibliography

Aki, K. and Richards, P. G. (2002). *Quantitative seismology*. University Science Books, Sausalito, Calif, 2nd edition.

Binder, G. (2018). Neural networks for moment-tensor Inversion of surface microseismic data. In *SEG Technical Program Expanded Abstracts 2018*, pages 2917–2921, Anaheim, California. Society of Exploration Geophysicists.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Information science and statistics. Springer, New York.

Biswas, R., Vassiliou, A., Stromberg, R., and Sen, M. K. (2018). Stacking velocity estimation using recurrent neural network. In *SEG Technical Program Expanded Abstracts 2018*, pages 2241–2245, Anaheim, California. Society of Exploration Geophysicists.

Biswas, R., Vassiliou, A., Stromberg, R., and Sen, M. K. (2019). Estimating normal moveout velocity using the recurrent neural network. *Interpretation*, 7(4):T819–T827.

Bouchon, M. (1981). A simple method to calculate Green's functions for elastic layered media. *Bulletin of the Seismological Society of America*, 71(4):959–971.

Bouchon, M. (2003). A review of the discrete wavenumber method. *Pure and Applied Geophysics*, 160:445–465.

Bouchon, M. and Aki, K. (1977). Discrete wave-number representation of seismic-source wave fields. *Bulletin of the Seismological Society of America*, 67(2):259–277.

Carrizo Mascarell, M. (2020). Extracting geomechanical information from borehole microseismic data with machine learning. Master's thesis, Delft University of Technology.

Cervený, V. (2001). *Seismic ray theory*. Cambridge University Press, Cambridge, U.K. ; New York.

Collins, D. S. (2000). Lithological Controls on Seismicity in Granitic Rocks. *Bulletin of the Seismological Society of America*, 90(3):709–723.

Cotton, F. and Coutant, O. (1997). Dynamic stress variations due to shear faults in a plane-layered medium. *Geophysical Journal International*, 128(3):676–688.

Dahm, T. and Krüger, F. (2014). Moment tensor inversion and moment tensor interpretation. In *New Manual of Seismological Observatory Practice 2 (NMSOP-2)*, pages 1–37. Deutsches GeoForschungsZentrum GFZ.

Das, S., Hobson, M. P., Feroz, F., Chen, X., Phadke, S., Goudswaard, J., and Hohl, D. (2021). Microseismic event detection in large heterogeneous velocity models using Bayesian multimodal nested sampling. *Data-Centric Engineering*, 2:e1.

Dayhoff, J. E. and DeLeo, J. M. (2001). Artificial neural networks: opening the black box. *Cancer*, 91(8 Suppl):1615–1635.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12(61):2121–2159.

Eyre, T. S. and van der Baan, M. (2015). Overview of moment-tensor inversion of microseismic events. *The Leading Edge*, 34(8):882–888.

Eyre, T. S. and van der Baan, M. (2017). The reliability of microseismic moment-tensor solutions: Surface versus borehole monitoring. *Geophysics*, 82(6):KS113–KS125.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Hansen, T. M., Cordua, K. S., Jacobsen, B. H., and Mosegaard, K. (2014). Accounting for imperfect forward modeling in geophysical inverse problems — Exemplified for crosshole tomography. *Geophysics*, 79(3):H1–H21.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv:1207.0580 [cs]*. arXiv: 1207.0580.

Huang, W., Wang, R., Li, H., and Chen, Y. (2017). Unveiling the signals from extremely noisy microseismic data for high-resolution hydraulic fracturing monitoring. *Scientific Reports*, 7(1):11996.

Häring, M. O., Schanz, U., Ladner, F., and Dyer, B. C. (2008). Characterisation of the Basel 1 enhanced geothermal system. *Geothermics*, 37(5):469–495.

Iqbal, N., Liu, E., McClellan, J. H., Al-Shuhail, A., Kaka, S. I., and Zerguine, A. (2018). Detection and Denoising of Microseismic Events Using Time–Frequency Representation and Tensor Decomposition. *IEEE Access*, 6:22993–23006.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer.

Jost, M. L. and Herrmann, R. B. (1989). A Student's Guide to and Review of Moment Tensors. *Seismological Research Letters*, 60(2):37–57.

Kanamori, H. (1977). The energy release in great earthquakes. *Journal of Geophysical Research*, 82(20):2981–2987.

Kennett, B. and Kerry, N. (1979). Seismic waves in a stratified half space. *Geophysical Journal International*, 57:557–583.

Kuang, W., Yuan, C., and Zhang, J. (2021). Real-time determination of earthquake focal mechanism via deep learning. *Nature Communications*, 12(1):1432.

Li, H., Chang, X., Xie, X.-B., and Wang, Y. (2021). Microseismic moment-tensor inversion and sensitivity analysis in vertically transverse isotropic media. *Geophysics*, 86(2):KS23–KS36.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133.

McGarr, A. (2003). Maximum Slip in Earthquake Fault Zones, Apparent Stress, and Stick-Slip Friction. *Bulletin of the Seismological Society of America*, 93(6):2355–2362.

Mustać, M. and Tkalčić, H. (2016). Point source moment tensor inversion through a Bayesian hierarchical model. *Geophysical Journal International*, 204(1):311–323.

Nolen-Hoeksema, R. C. and Ruff, L. J. (2001). Moment tensor inversion of microseisms from the B-sand propped hydrofracture, M-site, Colorado. *Tectonophysics*, 336(1-4):163–181.

Oristaglio, M. (2015). SEAM Update: The Arid Model — Seismic exploration in desert terrains. *The Leading Edge*, 34(4):466–468.

Ovcharenko, O., Akram, J., and Peter, D. (2018). Feasibility of moment tensor inversion from a single borehole data using artificial neural networks. GEO Bahrain 2018.

Oye, V., Aker, E., Daley, T. M., Kühn, D., Bohloli, B., and Korneev, V. (2013). Microseismic Monitoring and Interpretation of Injection Data from the in Salah CO2 Storage Site (Krechba), Algeria. *Energy Procedia*, 37:4191–4198.

Perol, T., Gharbi, M., and Denolle, M. (2018). Convolutional neural network for earthquake detection and location. *Science Advances*, 4(2):e1700578.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.

Sambridge, M. and Gallagher, K. (1993). Earthquake hypocenter location using genetic algorithms. *Bulletin of the Seismological Society of America*, 83(5):1467–1491.

Shapiro, S. A. (2015). *Fluid-induced seismicity*. Cambridge University Press, Cambridge.

Shearer, P. M. (2009). *Introduction to seismology*. Cambridge University Press, Cambridge. OCLC: 489719805.

Sipkin, S. A. (1986). Interpretation of non-double-couple earthquake mechanisms derived from moment tensor inversion. *Journal of Geophysical Research*, 91(B1):531.

Song, F. and Toksöz, M. N. (2011). Full-waveform based complete moment tensor inversion and source parameter estimation from downhole microseismic data for hydrofracture monitoring. *Geophysics*, 76(6):WC103–WC116.

Stein, S. and Wysession, M. (2005). *An introduction to seismology, earthquakes, and earth structure*. Blackwell, Malden, Mass. OCLC: 255467767.

Sun, J., Niu, Z., Innanen, K. A., Li, J., and Trad, D. O. (2020). A theory-guided deep-learning formulation and optimization of seismic waveform inversion. *Geophysics*, 85(2):R87–R99.

Sun, J., Wang, L., and Hou, H. (2012). Application of micro-seismic monitoring technology in mining engineering. *International Journal of Mining Science and Technology*, 22(1):79–83.

Tano, M. E., Portwood, G. D., and Ragusa, J. C. (2020). Accelerating Training in Artificial Neural Networks with Dynamic Mode Decomposition. *arXiv:2006.14371 [physics]*. arXiv: 2006.14371.

Tarantola, A. (2005). *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial and Applied Mathematics.

van der Baan, M. v., Eaton, D., and Dusseault, M. (2013). Microseismic Monitoring Developments in Hydraulic Fracture Stimulation. In Jeffrey, R., editor, *Effective and Sustainable Hydraulic Fracturing*. InTech.

Vavryčuk, V. (2001). Inversion for parameters of tensile earthquakes. *Journal of Geophysical Research: Solid Earth*, 106(B8):16339–16355.

Vavryčuk, V. (2007). On the retrieval of moment tensors from borehole data. *Geophysical Prospecting*, 55(3):381–391.

Vavryčuk, V. (2011). Tensile earthquakes: Theory, modeling, and inversion. *Journal of Geophysical Research*, 116(B12):B12320.

Vavryčuk, V. (2015). Inversion for the Composite Moment Tensor. *Bulletin of the Seismological Society of America*, 105(6):3024–3035.

Vera Rodriguez, I., Gu, Y. J., and Sacchi, M. D. (2011). Resolution of Seismic-Moment Tensor Inversions from a Single Array of Receivers. *Bulletin of the Seismological Society of America*, 101(6):2634–2642.

Warpinski, N. (2009). Microseismic Monitoring: Inside and Out. *Journal of Petroleum Technology*, 61(11):80–85.

Zdanov, M. S. (2015). *Inverse theory and applications in geophysics*. Elsevier, Amsterdam Boston Heidelberg London New York Oxford Paris San Diego San Francisco Singapore Sydney Tokyo, second edition.

Zhou, H., Zhang, W., and Zhang, J. (2016). Downhole microseismic monitoring for low signal-to-noise ratio events. *Journal of Geophysics and Engineering*, 13(5):805–816.

Appendix A

Calculation of the Green's functions derivatives using the modified DWM

A-1 P-SV, and SH reflection/transmission coefficients

The seismic displacements due to the P, SV and SH potentials described in Chapter 2 are calculated following the steps described in Cotton and Coutant (1997). The following sections contain a summary of the theory presented in Carrizo Mascarell (2020).

Let's consider the case where P-SV waves strike a horizontal interface between two homogeneous layers from above and below. Each possible incident P-SV wave (from above or below), can generate outgoing P-SV waves, yielding a total of 16 reflection/transmission coefficients as shown in figures A-1a, A-1b, A-1c, and A-1d. These coefficients are described as follows:

- $\grave{P}\acute{P}$: $P \longrightarrow P$ reflection coefficient (above).
- $\grave{P}\acute{S}$: $P \longrightarrow SV$ reflection/conversion coefficient (above).
- $\grave{S}\acute{P}$: $SV \longrightarrow P$ reflection/conversion coefficient (above).
- $\grave{S}\acute{S}$: $SV \longrightarrow SV$ reflection coefficient (above).
- $\acute{P}\acute{P}$: $P \longrightarrow P$ transmission coefficient (below).
- $\acute{P}\acute{S}$: $P \longrightarrow SV$ transmission/conversion coefficient (below).
- $\grave{P}\grave{P}$: $P \longrightarrow P$ transmission coefficient (above).
- $\grave{P}\grave{S}$: $P \longrightarrow SV$ transmission/conversion coefficient (above).
- $\grave{S}\grave{P}$: $SV \longrightarrow P$ transmission/conversion coefficient (above).
- $\grave{S}\grave{S}$: $SV \longrightarrow SV$ transmission coefficient (above).
- $\acute{P}\grave{P}$: $P \longrightarrow P$ reflection coefficient (below).
- $\acute{P}\grave{S}$: $P \longrightarrow SV$ reflection/conversion coefficient (below).

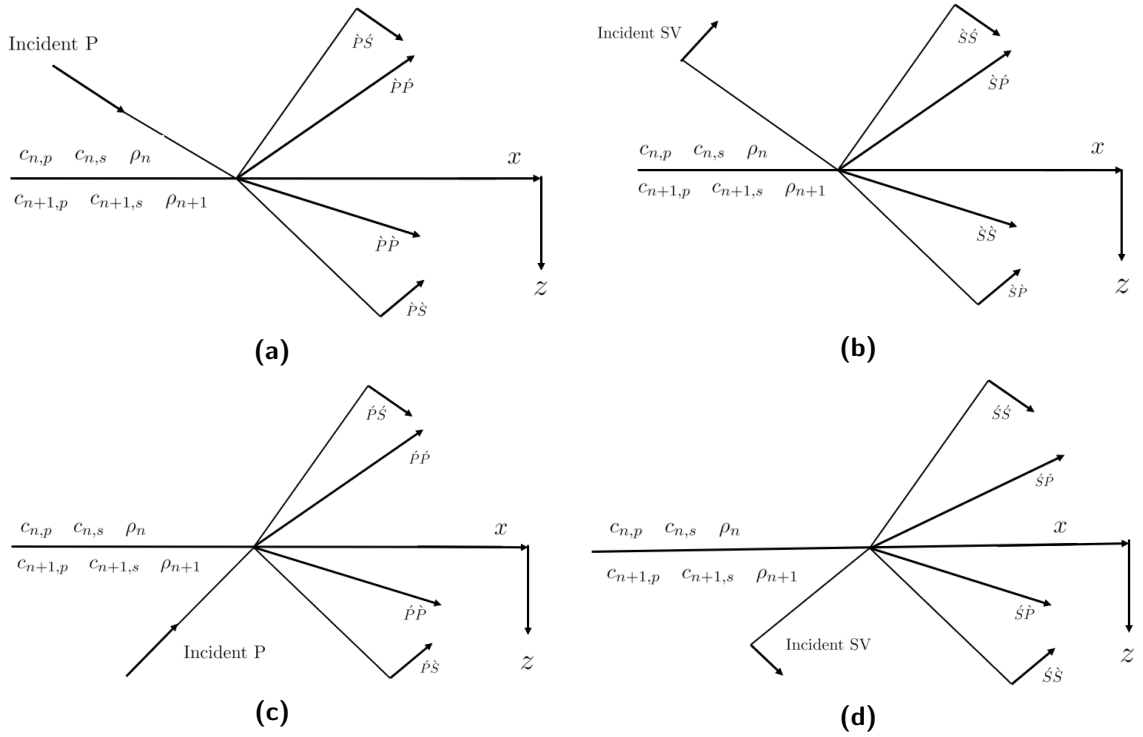


Figure A-1. Notation for the sixteen possible reflection and transmission coefficients of incident P-SV waves problems. The arrows in the figures denote the particle motion direction: (a), incident P-wave from above; (b), incident SV-wave from above; (c), incident P-wave from below; (d), incident SV-wave from below (modified from Aki and Richards (2002)).

- $\acute{S}\acute{P}$: $SV \longrightarrow P$ transmission/conversion coefficient (below).
- $\acute{S}\acute{S}$: $SV \longrightarrow SV$ transmission coefficient (below).
- $\acute{S}\grave{P}$: $SV \longrightarrow P$ reflection/conversion coefficient (below).
- $\acute{S}\grave{S}$: $SV \longrightarrow SV$ reflection coefficient (below).

In figures A-1a, A-1b, A-1c, and A-2b $c_{n,p}$, $c_{n,s}$, ρ_n are the P-wave velocity, S-wave velocity and the density value at the nth layer, respectively. Additionally, the magnitude of the different coefficients is given by the following equations:

$$\grave{P}\acute{P} = \left[\left(b\frac{\cos(i_n)}{c_{n,p}} - c\frac{\cos(i_{n+1})}{c_{n+1,p}}\right)F - \left(a + d\frac{\cos(i_n)}{c_{n,p}}\frac{\cos(j_{n+1})}{c_{n+1,s}}\right)Hp^2\right]/\mathcal{D}. \tag{A-1}$$

$$\grave{P}\acute{S} = -2\frac{\cos(i_n)}{c_{n,p}}\left(ab + cd\frac{\cos(i_{n+1})}{c_{n,p}}\frac{\cos_{j_{N+1}}}{c_{n+1,s}}\right)pc_{n,p}/(c_{n,s}\mathcal{D}). \tag{A-2}$$

$$\grave{P}\grave{P} = 2\rho_n\frac{\cos(i_n)}{c_{n,p}}Fc_{n,p}/(c_{n+1,p}\mathcal{D}). \tag{A-3}$$

$$\grave{P}\grave{S} = 2\rho_n\frac{\cos(i_n)}{c_{n,p}}Hpc_{n,p}/(c_{n+1,s}\mathcal{D}). \tag{A-4}$$

$$\grave{S}\acute{P} = -2\frac{\cos(j_n)}{c_{n,s}}\left(ab + cd\frac{\cos(i_{n+1})}{c_{n+1,p}}\frac{\cos(j_{n+1})}{c_{n+1,s}}\right)pc_{n,s}/(c_{n,p}\mathcal{D}). \tag{A-5}$$

$$\grave{S}\acute{S} = -\left[\left(b\frac{\cos(j_n)}{c_{n,s}} - c\frac{\cos(j_{n+1})}{c_{n+1,s}}\right)E - \left(a + d\frac{\cos(i_{n+1})}{c_{n+1,p}}\frac{\cos(j_n)}{c_{n,s}}\right)Gp^2\right]/\mathcal{D}. \tag{A-6}$$

$$\grave{S}\grave{P} = -2\rho_n\frac{\cos(j_n)}{c_{n,s}}Gpc_{n,s}/(c_{n+1,p}\mathcal{D}). \tag{A-7}$$

$$\grave{S}\grave{S} = 2\rho_n\frac{\cos(j_n)}{c_{n,s}}Ec_{n,s}/(c_{n+1,s}\mathcal{D}). \tag{A-8}$$

$$\acute{P}\acute{P} = 2\rho_{n+1}\frac{\cos(i_{n+1})}{c_{n+1,p}}Fc_{n+1,p}/(c_{n,p}\mathcal{D}). \tag{A-9}$$

$$\acute{P}\acute{S} = -2\rho_{n+1}\frac{\cos(i_{n+1})}{c_{n+1,p}}Gpc_{n+1,p}/(c_{n+1,p}\mathcal{D}). \tag{A-10}$$

$$\acute{P}\grave{P} = -\left[\left(b\frac{\cos(i_n)}{c_{n,p}} - c\frac{\cos(i_{n+1})}{c_{n+1,p}}\right)F + \left(a + d\frac{\cos(i_{n+1})}{c_{n+1,p}}\frac{\cos(j_n)}{c_{n,s}}\right)Gp^2\right]/\mathcal{D}. \tag{A-11}$$

$$\acute{P}\grave{S} = 2\frac{\cos(i_{n+1})}{c_{n+1,s}}\left(ac + bd\frac{\cos(i_n)}{c_{n,p}}\frac{\cos(j_n)}{c_{n,s}}\right)pc_{n+1,p}/(c_{n+1,s}\mathcal{D}). \tag{A-12}$$

$$\acute{S}\acute{P} = 2\rho_{n+1}\frac{\cos(j_{n+1})}{c_{n+1,s}}Hpc_{n+1,s}/(c_{n,s}\mathcal{D}). \tag{A-13}$$

$$\acute{S}\acute{S} = 2\rho_{n+1}\frac{\cos(j_{n+1})}{c_{n+1,s}}Ec_{n+1,s}/(c_{n,s}\mathcal{D}). \tag{A-14}$$

$$\acute{S}\grave{P} = 2\frac{\cos(j_{n+1})}{c_{n+1,s}}\left(ac + bd\frac{\cos(i_n)}{c_{n,p}}\frac{\cos(j_n)}{c_{n,s}}\right)pc_{n+1,s}/(c_{n+1,p}\mathcal{D}). \tag{A-15}$$

$$\acute{S}\grave{S} = \left[\left(b\frac{\cos(j_n)}{c_{n,s}} - c\frac{\cos(j_{n+1})}{c_{n+1,s}}\right)E + \left(a + d\frac{\cos(i_n)}{c_{n,p}}\frac{\cos(j_{n+1})}{c_{n+1,s}}\right)Hp^2\right]/\mathcal{D}. \tag{A-16}$$

$p = k/\omega$ is the *ray parameter* given the horizontal wavenumber k and the angular frequency ω and,

$$a = \rho_{n+1}(1 - 2c_{n+1,s}^2p^2) - \rho_n(1 - 2c_{n,s}^2p^2), \quad b = \rho_{n+1}(1 - 2c_{n+1,s}^2p^2) + 2\rho_nc_{n,s}^2p^2, \tag{A-17a}$$

$$c = \rho_n(1 - 2c_{n,s}^2p^2) + 2\rho_{n+1}c_{n+1,s}^2p^2, \quad d = 2(\rho_{n+1}c_{n+1,s}^2 - \rho_nc_{n,s}^2), \tag{A-17b}$$

$$E = b\frac{\cos(i_n)}{c_{n,p}} + c\frac{\cos(i_{n+1})}{c_{n+1,p}}, \quad F = b\frac{\cos(j_n)}{c_{n,s}} + c\frac{\cos(j_n)}{c_{n+1,s}}, \tag{A-18a}$$

$$G = a - d\frac{\cos(i_n)}{c_{n,p}}\frac{\cos(j_{n+1})s}{c_{n+1,s}}, \quad H = a - d\frac{\cos(i_{n+1})}{c_{n+1,p}}\frac{\cos(j_n)}{c_{n,s}}, \tag{A-18b}$$

$$\mathcal{D} = EF + GHp^2, \tag{A-19}$$

and the cosine terms can be written as follows:

$$\frac{\cos(i_n)}{c_{n,p}} = \sqrt{\frac{1}{c_{n,p}^2} - p^2}. \tag{A-20}$$

$$\frac{\cos(i_{n+1})}{c_{n+1,p}} = \sqrt{\frac{1}{c_{n+1,p}^2} - p^2}. \tag{A-21}$$

$$\frac{\cos(j_n)}{c_{n,s}} = \sqrt{\frac{1}{c_{n+1,s}^2} - p^2}. \tag{A-22}$$

$$\frac{\cos(j_{n+1})}{c_{n+1,s}} = \sqrt{\frac{1}{c_{n+1,s}^2} - p^2}. \tag{A-23}$$

The terms i_n and j_n are the angles orthogonal to the plane fronts at the nth layer for the P-waves and SV-waves, respectively (Aki and Richards, 2002). Similarly, a plane SH-wave that strikes the same interface between two homogeneous layers generates outgoing SH-waves (see figures A-2a and A-2b).

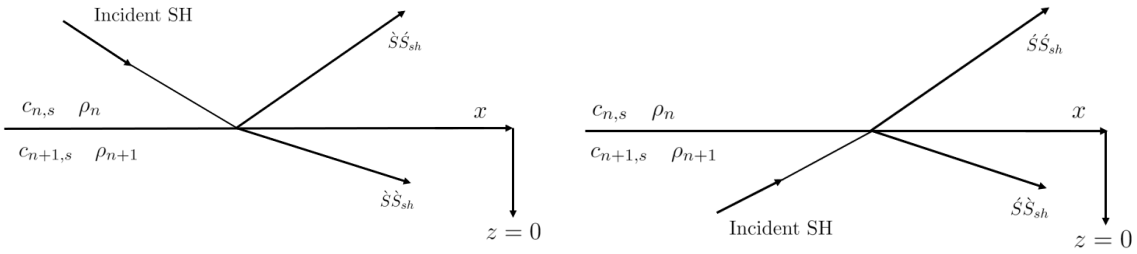


Figure A-2. Notation for the possible reflection and transmission coefficients of incident SH-waves. The arrows denote the particle motion direction: (a) incident SH-wave from above; (b), incident SH-wave from below (modified from Aki and Richards (2002)).

The reflection and transmission coefficients due to an incident SH-wave are given by the following relations:

$$\grave{S}\acute{S}_{sh} = \frac{\rho_n c_{n,s}\cos(j_n) - \rho_{n+1} c_{n+1,s}\cos(j_{n+1})}{\Delta}. \tag{A-24}$$

$$\acute{S}\acute{S}_{sh} = \frac{2\rho_{n+1} c_{n+1,s}\cos(j_{n+1})}{\Delta}. \tag{A-25}$$

$$\grave{S}\grave{S}_{sh} = \frac{2\rho_n c_{n,s}\cos(j_n)}{\Delta}. \tag{A-26}$$

$$\acute{S}\grave{S}_{sh} = -\grave{S}\acute{S}_{sh}. \tag{A-27}$$

$$\Delta = \rho_n c_{n,s} \cos(j_n) + \rho_{n+1} c_{n+1,s} \cos(j_{n+1}). \tag{A-28}$$

The reflection and transmission coefficients described for the P, SV, and SH waves can be summarized in matrix form for all possible conversion modes. This matrix is often referred to as the *scattering matrix* and for P-SV waves it can be written as follows:

$$\begin{pmatrix} \grave{P}\acute{P} & \grave{S}\acute{P} & \acute{P}\acute{P} & \acute{S}\acute{P} \\ \grave{P}\acute{S} & \grave{S}\acute{S} & \acute{P}\acute{S} & \acute{S}\acute{S} \\ \grave{P}\grave{P} & \grave{S}\grave{P} & \acute{P}\grave{P} & \acute{S}\grave{P} \\ \grave{P}\grave{S} & \grave{S}\grave{S} & \acute{P}\grave{S} & \acute{S}\grave{S} \end{pmatrix} = \begin{pmatrix} \mathbf{R}_u & \mathbf{T}_u \\ \mathbf{T}_d & \mathbf{R}_d \end{pmatrix}, \tag{A-29}$$

where

$$\mathbf{R}_u = \begin{pmatrix} \grave{P}\acute{P} & \grave{S}\acute{P} \\ \grave{P}\acute{S} & \grave{S}\acute{S} \end{pmatrix}, \tag{A-30}$$

$$\mathbf{T}_u = \begin{pmatrix} \acute{P}\acute{P} & \acute{S}\acute{P} \\ \acute{P}\acute{S} & \acute{S}\acute{S} \end{pmatrix}, \tag{A-31}$$

$$\mathbf{T}_d = \begin{pmatrix} \grave{P}\grave{P} & \grave{S}\grave{P} \\ \grave{P}\grave{S} & \grave{S}\grave{S} \end{pmatrix}, \tag{A-32}$$

$$\mathbf{R}_d = \begin{pmatrix} \acute{P}\grave{P} & \acute{S}\grave{P} \\ \acute{P}\grave{S} & \acute{S}\grave{S} \end{pmatrix}. \tag{A-33}$$

A similar matrix as the one defined for the P-SV waves can be defined to summarize the conversion modes for SH-waves as follows:

$$\begin{pmatrix} \grave{S}\acute{S}_{sh} & \acute{S}\acute{S}_{sh} \\ \grave{S}\grave{S}_{sh} & \acute{S}\grave{S}_{sh} \end{pmatrix} = \begin{pmatrix} R_{u_{sh}} & T_{u_{sh}} \\ T_{d_{sh}} & R_{d_{sh}} \end{pmatrix}. \tag{A-34}$$

The difference in equation (A-34) compared to equation (A-29) is that the obtained reflections and transmissions are scalars compared to the case of P-SV waves.

A-2 Calculation of the reflectivity and transmissivity matrices

Using the reflection and transmission matrices (for P-SV waves), it is possible to propagate a wavefield $\mathbf{v}$ from a depth level z_n to a level z_{n+1} through a linear transformation $\mathbf{A}(z_{n+1}, z_n)$ using equations (A-30), (A-31), (A-32) and (A-33):

$$\mathbf{v}(z_{n+1}) = \mathbf{A}(z_{n+1}, z_n)\mathbf{v}(z_n). \tag{A-35}$$

The linear transformation $\mathbf{A}$ can be expressed as follows:

$$\mathbf{A}(z_{n+1}, z_n) = \mathbf{L}_u^{-1}(z_n, z_{n+1})\mathbf{L}_l(z_{n+1}, z_n), \tag{A-36}$$

where

$$\mathbf{L}_u(z_n, z_{n+1}) = \begin{pmatrix} \mathbf{T}_u & \mathbf{0} \\ -\mathbf{R}_u & \mathbf{I} \end{pmatrix}, \quad \mathbf{L}_l(z_{n+1}, z_n) = \begin{pmatrix} \mathbf{I} & -\mathbf{R}_d \\ \mathbf{0} & \mathbf{T}_d \end{pmatrix}. \tag{A-37}$$

$\mathbf{I}$ is the identity matrix and $\mathbf{0}$ is a matrix filled with zeros. In Kennett and Kerry (1979) the wavefield potential $\mathbf{v}$ can also be propagated through a medium of n layers with different velocities and densities. For a stack of three layers ($z_1 < z < z_3$), the reflection and transmission matrices can be calculated as follows:

$$\begin{aligned}
\mathbf{R}_d(z_3, z_1) &= \mathbf{R}_d(z_2, z_1) + \mathbf{T}_u(z_1, z_2)\mathbf{R}_d(z_3, z_2)[\mathbf{I} - \mathbf{R}_u(z_1, z_2)\mathbf{R}_d(z_3, z_2)]^{-1}\mathbf{T}_d(z_2, z_1). \\
\mathbf{T}_d(z_3, z_1) &= \mathbf{T}_d(z_3, z_2)[\mathbf{I} - \mathbf{R}_u(z_1, z_2)\mathbf{R}_d(z_3, z_2)]^{-1}\mathbf{T}_d(z_2, z_1). \\
\mathbf{R}_u(z_1, z_3) &= \mathbf{R}_u(z_2, z_3) + \mathbf{T}_d(z_3, z_2)\mathbf{R}_u(z_1, z_2)[\mathbf{I} - \mathbf{R}_d(z_3, z_2)\mathbf{R}_u(z_1, z_2)]^{-1}\mathbf{T}_u(z_2, z_3). \\
\mathbf{T}_u(z_1, z_3) &= \mathbf{T}_u(z_1, z_2)[\mathbf{I} - \mathbf{R}_d(z_3, z_2)\mathbf{R}_u(z_1, z_2)]^{-1}\mathbf{T}_u(z_2, z_3).
\end{aligned} \tag{A-38}$$

A similar scheme can be applied using the reflection and transmission coefficients for SH-waves.

A-3 Source potentials

The potentials Φ, Ψ, and χ can radiate upwards or downwards from the source layer. Thus, a total of six source potentials are obtained. The derived expressions for the potentials radiated by the six independent moment-tensor sources are summarized in Table A-1.

Table A-1. Source potentials.

Wave mode	Source potential	Direction
P	$S_1(z) = e^{-ik_{z,p}\lvert z-z_0\rvert}$	Upward
P	$S_2(z) = \text{sign}(z-z_0)e^{-ik_{z,p}\lvert z-z_0\rvert}$	Downward
SV	$S_3(z) = e^{-ik_{z,s}\lvert z-z_0\rvert}$	Upward
SV	$S_4(z) = \text{sign}(z-z_0)e^{-ik_{z,s}\lvert z-z_0\rvert}$	Downward
SH	$S_5(z) = e^{-ik_{z,s}\lvert z-z_0\rvert}$	Upward
SH	$S_6(z) = \text{sign}(z-z_0)e^{-ik_{k,s}\lvert z-z_0\rvert}$	Downward

z_0 represents the source depth, $k_{z,p}$ and $k_{z,s}$ are the P-wave and S-wave vertical wavenumbers:

$$k_{z,p} = \sqrt{\frac{\omega^2}{c_p^2} - k_r^2}, \quad k_{z,s} = \sqrt{\frac{\omega^2}{c_s^2} - k_r^2}, \quad \text{with} \quad \text{Im}(k_{z,p}),\, \text{Im}(k_{z,s}) < 0, \tag{A-39}$$

$k_r = 2\pi/\Delta r$ is the radial wavenumber, Δr is the spatial period as defined in Bouchon (1981) and r is the distance from the source to the receiver layer. The six elementary sources shown in Table A-1 due to unidirectional forces are used to decompose the potentials Φ, Ψ, and χ for the six independent moment-tensor components as in Cotton and Coutant (1997):

$$
\begin{aligned}
M_{XY}: &\begin{cases} \Phi = C_f F\ \sin(2\theta)\, cs2\, K_1\, S_1 \\ \Psi = C_f F\ \sin(2\theta)\, K_1\, S_4 \\ \chi = -C_f F\ \cos(2\theta)\, cs3\, K_1\, S_5 \end{cases} \qquad
M_{XZ}: \begin{cases} \Phi = C_f F\, 2\cos(\theta)(-k_r^2)\, J_1\, S_2 \\ \Psi = C_f F\ \cos(\theta)\, cs9\, J1\, S_3 \\ \chi = C_f F\ \sin(\theta)(-k_\beta^2)\, J_1\, S_6 \end{cases} \\
M_{YZ}: &\begin{cases} \Phi = C_f F\, 2\sin(\theta)(-k_r^2)\, J_1\, S_2 \\ \Psi = C_f F\ \sin(\theta)\, cs9\, J_1\, S_3 \\ \chi = C_f F\ \cos(\theta)\, k_\beta^2\, J_1\, S_3 \end{cases} \qquad
M_{XX}: \begin{cases} \Phi = C_f F\, cs2\, (\cos^2(\theta)\, K_1 + K_2))S_1 \\ \Psi = C_f F\, (\cos^2(\theta)\, K_1 + K_2)S_4 \\ \chi = C_f F\ \cos(\theta)\sin(\theta)\, cs3\, K_1\, S_5 \end{cases} \\
M_{YY}: &\begin{cases} \Phi = C_f F\, cs2(\sin^2(\theta)\, K_1 + K_2)S_1 \\ \Psi = C_f F\, (\sin^2(\theta)\, K_1 + K_2)S_4 \\ \chi = -C_f\ \cos(\theta)\sin(\theta)\, cs3\, K_1\, S_5 \end{cases} \qquad
M_{ZZ}: \begin{cases} \Phi = C_f F\, (-ik_{z,p}k_r)\, J_0\, S_1 \\ \Psi = -C_f F\, J_0\, S_4 \\ \chi = 0 \end{cases}
\end{aligned}
\tag{A-40}
$$

where

$$
\begin{aligned}
K_0 &= k_r J_0(k_r r), \\
K_1 &= k_r J_0(k_r r) - 2J_1(k_r r), \\
K_2 &= J_1(k_r r)/r, \\
K_3 &= -\frac{2}{r}K_1 - k_r^2 J_1(k_r r), \\
K_4 &= \frac{K_1}{r}, \\
K_5 &= k_r J_0(k_r r) - J_1(k_r r)/r,
\end{aligned}
\tag{A-41}
$$

and

$$
\begin{aligned}
C_f &= \frac{1}{2\rho\omega^2\Delta r}, \\
k_\alpha &= \frac{\omega}{c_p},\ k_\beta = \frac{\omega}{c_s}, \\
cs2 &= \frac{k_r^2}{ik_{z,p}},\ cs3 = \frac{k_\beta^2}{ik_{z,s}},\ cs9 = \frac{k_\beta^2 - 2k_r^2}{ik_{z,s}}.
\end{aligned}
\tag{A-42}
$$

A-4 Propagation of the six elementary sources

The six elementary sources can radiate upwards and downwards potentials from the source layer through the medium. Thus, for every potential Φ, Ψ or χ the following upgoing and downgoing components can be obtained:

$$
\begin{aligned}
S_1 &\longrightarrow S_1^{\Phi\, up}, S_1^{\Phi\, down}, S_1^{\Psi\, up}, S_1^{\Psi\, down},\\
S_2 &\longrightarrow S_2^{\Phi\, up}, S_2^{\Phi\, down}, S_2^{\Psi\, up}, S_2^{\Psi\, down},\\
S_3 &\longrightarrow S_3^{\Phi\, up}, S_3^{\Phi\, down}, S_3^{\Psi\, up}, S_3^{\Psi\, down},\\
S_4 &\longrightarrow S_4^{\Phi\, up}, S_4^{\Phi\, down}, S_4^{\Psi\, up}, S_4^{\Psi\, down},\\
S_5 &\longrightarrow S_5^{\chi\, up}, S_5^{\chi\, down},\\
S_6 &\longrightarrow S_6^{\chi\, up}, S_6^{\chi\, down},
\end{aligned}
\tag{A-43}
$$

where the symbol $\longrightarrow$ represents the conversion of the incident potential wavefield at the source layer. The different potentials propagate upwards or downwards for the six elementary sources using the reflection and transmission matrices described in Section A-1:

$$
\begin{aligned}
S_n^{\phi\, up} &= P_n^{\phi\, up}\, e^{ik_{z,p}(z-z_i)}, \quad S_n^{\phi\, down} = P_n^{\phi\, down}\, e^{-ik_{z,p}(z-z_i)},\\
S_n^{\psi\, up} &= P_n^{\psi\, up}\, e^{ik_{z,s}(z-z_i)}, \quad S_n^{\psi\, down} = P_n^{\psi\, down}\, e^{-ik_{z,s}(z-z_i)},\\
S_n^{\chi\, up} &= P_n^{\chi\, up}\, e^{ik_{z,s}(z-z_i)}, \quad S_n^{\chi\, down} = P_n^{\chi\, down}\, e^{-ik_{z,s}(z-z_i)}.
\end{aligned}
\tag{A-44}
$$

P_n^m are the reflection or transmission coefficients where m can be Φ, Ψ or χ. For P-SV waves the reflections and transmissions are 2×2 matrices, whereas for SH-waves are scalars.

A-5 Calculation of the seismic displacements

The spatial derivatives of Φ, Ψ, and χ in cylindrical coordinates are calculated to obtain the seismic displacements for every elementary source:

$$
u_r = \frac{\partial \Phi}{\partial r} + \frac{\partial^2 \Psi}{\partial r \partial z} + \frac{1}{r}\frac{\partial \chi}{\partial \theta}, \tag{A-45}
$$

$$
u_\theta = \frac{1}{r}\frac{\partial \Phi}{\partial \theta} + \frac{1}{r}\frac{\partial^2 \Psi}{\partial \theta \partial z} - \frac{\partial \chi}{\partial r}, \tag{A-46}
$$

$$
u_z = \frac{\partial \Phi}{\partial z} - \frac{\partial^2 \Psi}{\partial r^2} - \frac{1}{r}\frac{\partial \Psi}{\partial r} - \frac{1}{r^2}\frac{\partial^2 \Psi}{\partial \theta^2}. \tag{A-47}
$$

After calculating the partial derivatives, a total of 18 seismic displacements are obtained:

$$
M_{XY}: \begin{cases} u_r = C_f \sin(2\theta)\,(K_3[cs2(S_1^\phi + \epsilon i k_{z,p} S_1^\psi) + (S_4^\phi + \epsilon i k_{z,p} S_4^\psi)] + 2cs3\, K_4\, S_5^\chi) \\ u_\theta = C_f \cos(2\theta)\,(2K_4[cs2(S_1^\phi + \epsilon k_{z,p} S_4^\psi) + (S_4^\phi + \epsilon i k_{z,p} S_4^\phi)] + cs3\, K_3 S_5^\chi) \\ u_z = C_f \sin(2\theta)\, K_1[cs2(\epsilon i k_{z,s} S_1^\phi + k_r^2 S_1^\psi) + (\epsilon i k_{z,s} S_4^\phi + k_r^2 S_4^\psi)] \end{cases}
$$

$$
M_{XZ}: \begin{cases} u_r = C_f \cos(\theta)\,(K_5[-2k_r^2(S_2^\phi + \epsilon i k_{z,p} S_2^\psi) + cs9(S_3^\phi + \epsilon i k_{z,p} S_3^\psi)] - k_\beta^2 K_2 S_6^\chi) \\ u_\theta = C_f \sin(\theta)\,(K_2[2k_r^2(S_2^\phi + \epsilon i k_{z,p} a S_2^\psi) - + k_\beta^2 K_5 S_6^\chi]) \\ u_z = C_f \cos(\theta)\, J_1[-2k_r^2(\epsilon i k_{z,s} S_2^\phi + k_r^2 S_2^\psi) + cs9(\epsilon i k_{z,s} S_3^\phi + k_r^2 S_3^\psi)] \end{cases}
$$

$$
M_{YZ}: \begin{cases} u_r = C_f \sin(\theta)\,(K_5[-2k_r^2(S_2^\phi + \epsilon i k_{z,p} S_2^\psi) + cs9(S_3^\phi + \epsilon i k_{z,p} S_3^\psi)] - k_\beta^2 K_2 S_6^\chi) \\ u_\theta = C_f \cos(\theta)\,(K_2[-2k_r^2(S_2^\phi + \epsilon i k_{z,p} S_2^\psi) + cs9(S_3^\phi + \epsilon i k_{z,p} S_3^\psi)] - k_\beta^2 K_5 S_6^\chi) \\ u_z = C_f \sin(\theta)\, J_1[-2k_r^2(\epsilon i k_{z,s} S_2^\phi + k_r^2 S_2^\psi) + cs9(\epsilon i k_{z,s} S_3^\phi + k_r^2 S_3^\psi)] \end{cases}
$$

$$
M_{XX}: \begin{cases} u_r = C_f\,((K_3 \cos^2(\theta) + K_4)[cs2(S_1^\phi + \epsilon i k_{z,p} S_1^\psi) + (S_4^\phi + \epsilon i k_{z,p} S_4^\psi)] + \cos(2\theta)\, cs3\, K_4\, S_5^\chi) \\ u_\theta = -C_f\, \sin(2\theta)(K_4[cs2(S_1^\phi + \epsilon i k_{z,p} S_1^\psi) + (S_4^\phi + \epsilon i k_{z,p} S_4^\psi)] + \frac{cs3}{2}\, K_3\, S_5^\chi) \\ u_z = C_f(K_1 \cos^2(\theta) + K_2)[cs2(\epsilon i k_{z,s} S_1^\phi k_r^2 S_1^\psi) + (\epsilon i k_{z,s} S_4^\phi + k_r^2 S_4^\psi)] \end{cases}
$$

$$
M_{YY}: \begin{cases} u_r = C_f\,((K_3 \sin^2(\theta) + K_4)[cs2(S_1^\phi + \epsilon i k_{z,p} S_1^\psi) + (S_4^\phi + \epsilon i k_{z,p} S_4^\psi)] - \cos(2\theta)\, cs3\, K_4\, S_5^\chi) \\ u_\theta = C_f \sin(2\theta)\,(K_4[cs2(S_1^\phi + \epsilon i k_{z,p} S_1^\psi) + (S_4^\phi + \epsilon i k_{z,p} S_4^\psi)] + \frac{cs3}{2}\, K_3\, S_5^\chi) \\ u_z = C_f\,(K_1 \sin^2(\theta) + K_2)[cs2(\epsilon i k_{z,s} S_1^\phi + k_r^2 S_1^\psi) + (\epsilon i k_{z,s} S_4^\phi + k_r^2 S_4^\psi)] \end{cases}
$$

$$
M_{ZZ}: \begin{cases} u_r = C_f\, J_1\, k_r^2[i k_{z,s}(S_1^\phi + \epsilon i k_{z,p} S_1^\psi) + (S_4^\phi + \epsilon i k_{z,p} S_4^\psi)] \\ u_\theta = 0 \\ u_z = -C_f\, J_0\, k_r[i k_{z,s}(\epsilon i k_{z,s} S_1^\phi + k_r^2 S_1^\psi) + (\epsilon i k_{z,s} S_4^\phi + k_r^2 S_4^\psi)] \end{cases}
\tag{A-48}
$$

In the previous equations ϵ can be 1 for upgoing waves, or -1 for downgoing waves.