# 3D velocity-depth model building using surface seismic and well data

Paul Sexton

**How to cite:**

Sexton, Paul (1998) 3D velocity-depth model building using surface seismic and well data. Doctoral thesis, Durham University.

# 3D Velocity-Depth Model Building using Surface Seismic and Well Data

## Paul Sexton

**A thesis submitted in partial fulfilment
of the requirements for the degree of
Doctor of Philosophy**

**Department of Geological Sciences
The University of Durham
1998**

# 3D Velocity-Depth Model Building using Surface Seismic and Well Data

## Abstract

The objective of this work was to develop techniques that could be used to rapidly build a three-dimensional velocity-depth model of the subsurface, using the widest possible variety of data available from conventional seismic processing and allowing for moderate structural complexity. The result is a fully implemented inversion methodology that has been applied successfully to a large number of diverse case studies.

A model-based inversion technique is presented and shown to be significantly more accurate than the analytical methods of velocity determination that dominate industrial practice. The inversion itself is based around two stages of ray-tracing. The first takes picked interpretations in migrated-time and maps them into depth using a hypothetical interval velocity field; the second checks the validity of this field by simulating fully the kinematics of seismic acquisition and processing as accurately as possible. Inconsistencies between the actual and the modelled data can then be used to update the interval velocity field using a conventional linear scheme.

In order to produce a velocity-depth model that ties the wells, the inversion must include anisotropy. Moreover, a strong correlation between anisotropy and lithology is found. Unfortunately, surface seismic and well-tie data are not usually sufficient to uniquely resolve all the anisotropy parameters; however, the degree of non-uniqueness can be measured quantitatively by a resolution matrix which demonstrates that the model parameter trade-offs are highly dependent on the model and the seismic acquisition. The model parameters are further constrained by introducing well seismic traveltimes into the inversion. These introduce a greater range of propagation angles and reduce the non-uniqueness.

## Acknowledgements

This work was carried out at the Elf Geoscience Research Centre in London within the framework of a project on anisotropy. Many of the concepts expressed in this thesis are the culmination of many peoples ideas and I would like to thank everyone at Elf for their help, especially Etienne Robein and Paul Williamson, who have both taught me many things, including some geophysics. I am very grateful to Peter Swaby who helped me with the software development and transformed me from someone who thought he could program a computer into someone who could. Thanks also to Neil Goulty, my PhD supervisor at Durham for his advice and characteristic diligence. I must also thank Elf Aquitaine Production and Elf UK for the funding that made this work possible and for permission to publish the results. Finally, thanks to my wife, Juliet, for her support and endless patience.

# Table of Contents

# List of Figures

# 1.0 Introduction

## 1.1 Overview of the Inversion Procedure

The inversion technique is designed to replace the standard Dix-type inversion schemes (Dix, 1955) which currently dominate industrial practice. Al-Chalabi (1994) detailed the many approximations used in these standard analytical methods of velocity estimation. Velocity heterogeneity, finite offsets and anisotropy all contribute to a systematic mismatch between stacking velocity and the vertical root-mean square (RMS) velocity, which are assumed to be identical. In this thesis I present a model-based technique which allows many of these fundamental problems to be overcome successfully.

The inversion is based around two stages of ray-tracing. The first of these takes the picked interpretations in migrated-time and maps them into depth using a hypothetical interval velocity field; the second checks the validity of this field by simulating as fully as possible the kinematics of seismic acquisition and processing. Kinematic in this context means that we are only interested in traveltimes and not in the amplitude or shape of the seismic wavelet itself. Specifically, the stacking velocity field, the depth of the interpreted surfaces at the wells, and the well seismic traveltimes are modelled. Inconsistencies between the actual and the modelled data can then be used to update the interval velocity field and the process iterated until convergence. A schematic showing the processing sequence is shown in figure 1-1.

Primarily, the input data consist of:

- picked interpretations in migrated time,
- survey design parameters,
- full history of the processing including the time-migration velocity model,
- stacking velocity field after processing,
- well trajectories with interpretation markers identified, and
- picked well-seismic traveltimes.

The output is a three-dimensional (3D) velocity model and the interpreted surfaces mapped into depth. The velocity-depth model is parameterised by these layer boundaries and an analytical velocity parameterisation within each layer. The inversion procedure can be applied in a layer-stripping technique in which the velocity parameters of the shallowest layer are inverted and then fixed before proceeding to the next layer. Alternatively, the model parameters within several layers can be inverted for at once.

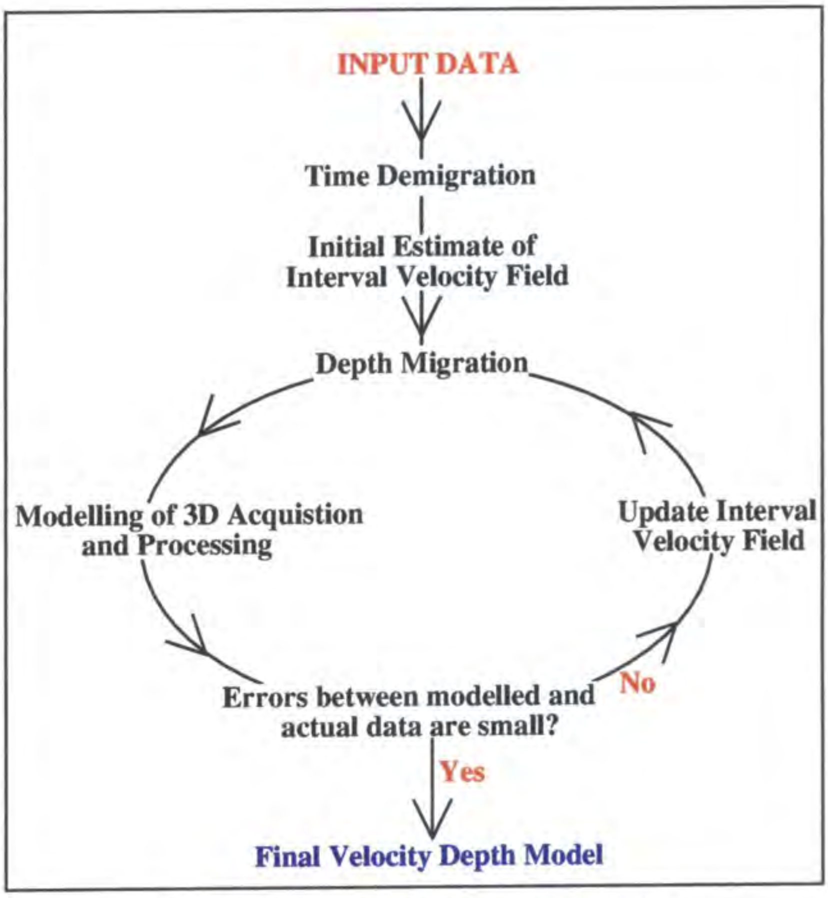
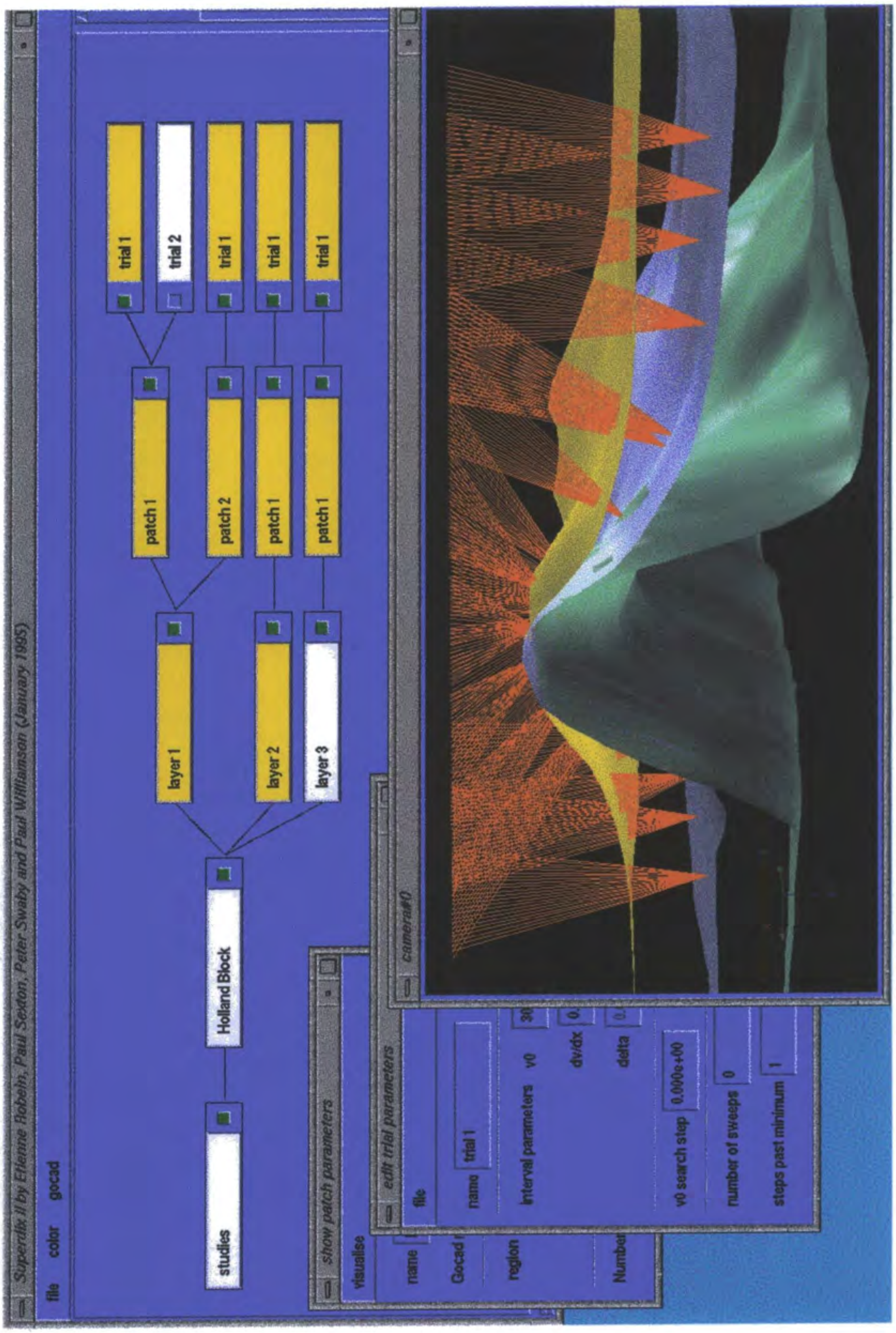


*Figure 1-1: Overview of the inversion procedure*

The project has required the development of a large source code amounting to some 45,000 lines (around 700 A4 pages) of C. A great deal of care was devoted to defining the underlying structure of the application that has been created. This effort, although initially time-consuming, has proved worthwhile as new ideas and algorithms were subsequently added with little overhead, and the research progressed very much quicker.



name trial 1

v0 search step 0.000e+00

number of sweeps 0

steps past minimum 1

*Figure 1-2: Screen display showing the inversion software platform that has been developed*

A user interface has been designed, and implemented, to make the functionality as easily accessible as possible (figure 1-2). The algorithms have been designed to be robust enough to be used industrially, and many practical issues have been addressed. More generally, the project has produced a geophysical development platform that allows new research ideas to be tested quickly by providing a flexible set of tools for kinematic modelling of data and mapping of data between the different seismic domains. The resultant application also provides a very quick and easy way of checking the sensitivity of acquisition and processing parameters to the data and how these parameters affect the resolution of the model parameters during subsequent inversion.

## 1.2 Historical Development of the Project

The project began with the aim of producing a stacking velocity inversion tool, an early prototype of which existed within Elf before the thesis began (Bernard Raynaud and Pierre Thore, 1994, personal communication). Early trials of this tool proved it not to be flexible enough to deal with real datasets or moderately complex structure. In this prototype application, the inversion was very simple and only consisted of a scan over a set of constant interval velocities in order to minimize the average error between the modelled and actual stacking velocities; and there was no way of incorporating well information. Despite this, the approach was potentially promising with some very interesting and original concepts, such as the simulation of the DMO operator during the modelling of the stacking velocities (Robein et al., 1995). It was decided that a new tool should be developed that took the best ideas from the prototype and refined and extended them in order to produce a useful operational tool. In the first stage of my work, a new stacking velocity inversion tool was developed from scratch. The model parameterisation and inversion methodology were changed substantially from the prototype and original algorithms developed for the domain mapping and ray-tracing. These improvements demonstrated that the approach was stable enough to be used with real data.

Stacking velocity inversion is not a concept original to this thesis (Hadley et al., 1988, Chiu and Stewart, 1987). In addition, there are several commercially available tools to perform stacking velocity inversion: GeoDepth[1] and TOMCAD[2] are examples of tools that include such functionality. The work presented here, however, does contains many original aspects, with significant differences in all aspects of the inversion procedure.

After the development of this new stacking velocity inversion tool, the velocity-depth models produced were compared with the depths from the well tops. In some cases, particularly in the North Sea, a large discrepancy was noticed. The depths from the stacking velocity inversion were, in places, found to be more than 10% deeper than those predicted from the wells (figure 1-3). This mistie between surface seismic inversion and the well logs had already been noted (Banik, 1984) and attributed to anisotropy (the variation of propagation velocity with direction) in the overburden, which strongly correlated with thick shale intervals.

To produce a single model that satisfies both the stacking velocities and the well depths, the inversion was extended to allow for anisotropy. The algorithm was rewritten to jointly invert for the well marker depths as well as the stacking velocities to produce an anisotropic parameterisation of the velocity model. With a increased number of model parameters the simple scanning technique, used previously, was no longer practical and was replaced with a least-squares optimisation procedure. The problem with this scheme, however, is that although a model can be generated that ties both the wells and the stacking velocities (figure 1-3), the anisotropic parameters cannot usually be uniquely resolved from surface seismic alone. I have studied this non-uniqueness

---

[1] GeoDepth is a trademark of Paradigm Geophysical

[2] TOMCAD is a trademark of Petrosystems - a CGG company

through construction of a resolution matrix and it has yielded important insights into anisotropic velocity inversion.

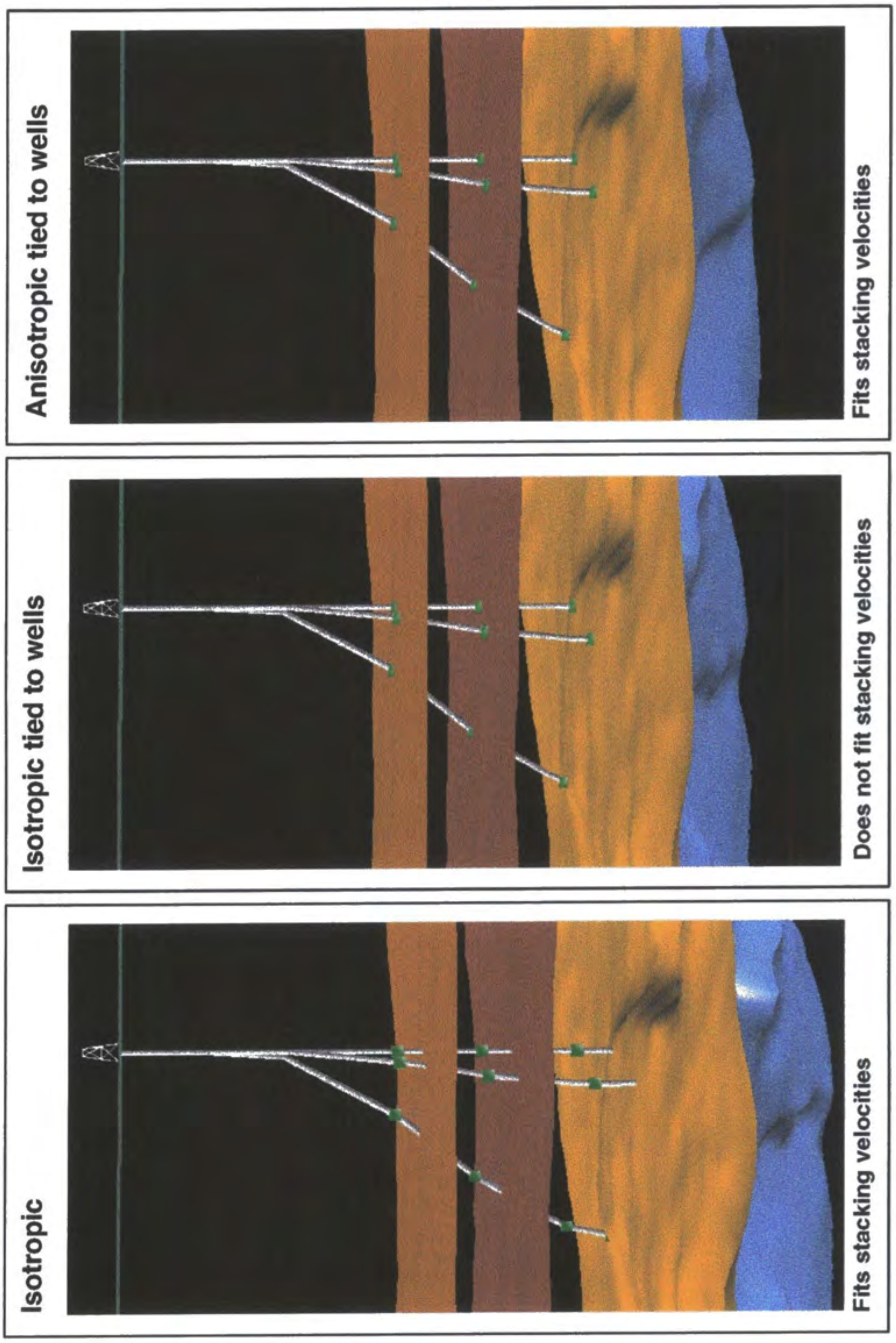


*Figure 1-3: Comparison of model building schemes*

In an attempt to further constrain the anisotropy, well seismic traveltimes were added as an extra piece of input data. These provide a greater range of propagation angles than the surface seismic data alone and help to resolve the non-uniqueness of the inversion result. The addition of well seismic traveltimes, however, means that a layer stripping approach is no longer suitable and so the inversion scheme was extended to invert for all the model parameters at once, in what we term the multi-layer joint inversion.

## 1.3 Synopsis

Chapter 2 describes the mapping of interpreted data between the stack, migrated-time and migrated-depth domains, which is essential for the inversion process. The mapping procedures are described in the context of the inversion and demonstrated using both synthetic and real data examples.

The forward problem of modelling the post-processed data is discussed in Chapter 3. The inverse problem is outlined in Chapter 4 with an emphasis on the non-uniqueness of the model parameter estimation. Chapter 5 consists of a complete case history using a real 3D dataset, and Chapter 6 presents the conclusions of the research as well as suggesting proposals for future work.

# 2.0 Domain Mapping

## 2.1 Overview

There are three seismic domains that must be considered during the inversion: the stack-domain, the migrated-time domain and the migrated-depth domain. They are related to each other as shown in figure 2-1. After a brief description of each, this chapter explains the transformation of interpreted data between the domains. The first stage of the inversion is to convert the migrated-time interpretations into migrated-depth. This chapter demonstrates that direct mapping between the time- and depth-migrated domains makes commonly invalid assumptions and therefore it is better to perform this mapping by first transforming into the stack domain.

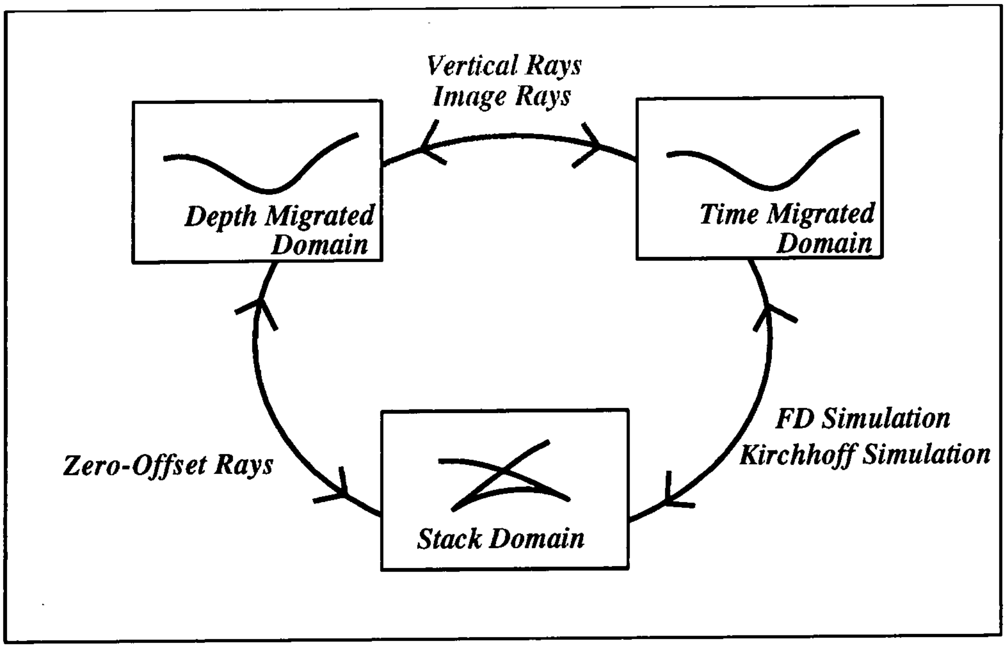


*Figure 2-1: The three seismic domains and their relationship to each other*

I also introduce an important original concept that uses the connectivity of data within, and between, the domains to ensure the consistency of the mapping and greatly improve its efficiency and stability. Finally, the accuracy of the mapping is demonstrated on a variety of real and synthetic datasets.

## 2.2 The Seismic Domains

### 2.2.1 The Stack Domain

Basic seismic processing begins by sorting all the traces which have the same source-receiver midpoint and assigning them to a common midpoint (CMP) gather. Figure 2-2 shows an example of a CMP gather for a single horizontal reflector with a constant velocity isotropic overburden. When the traces in the gather are sorted by source-receiver offset, $h$, the reflection produces a coherent event. In this simple case, the reflection points are all at the same location (the CMP principle), and the reflected event lies along a hyperbola which can be parameterised by the zero-offset traveltime, $t_0$, and the propagation velocity, $V$, within the layer.

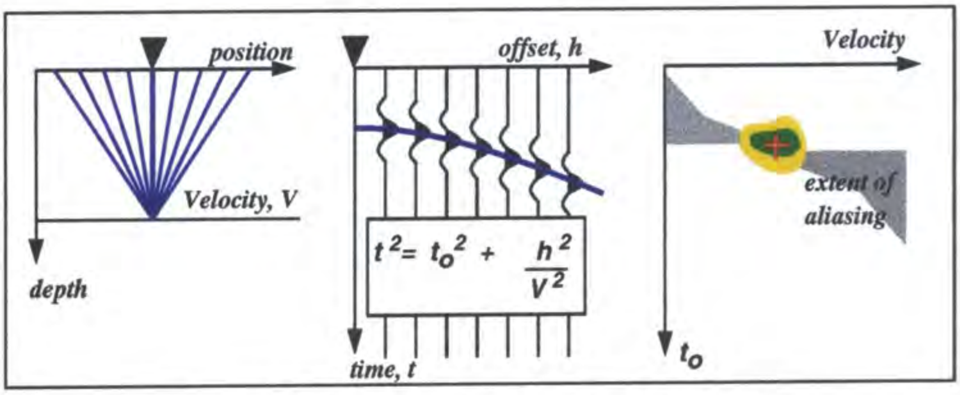


*Figure 2-2: Traveltime-offset relationship for 1D earth*

In seismic processing these gathers are routinely analysed by maximising a semblance function (Neidell and Taner, 1971). Traces belonging to a common midpoint are summed along hyperbolic trajectories defined by a variety of velocities and zero-offset traveltimes and the one that produces the highest energy of summation is selected. A semblance plot is shown on the right of figure 2-2. Each point on this plot represents a different hyperbola and is coloured depending on the summation energy along it. Due to the finite bandwidth of the seismic signal, there is an area of aliasing that makes a distinctive 'butterfly' pattern on the semblance plot and the semblance peak itself can have a significant width, especially in velocity. Often the intersection of the lines that define the extent of the aliasing are used to help guide the picking.

As the earth gets more complicated (figure 2-3), the traveltime-offset curve is no longer exactly hyperbolic. The common-midpoint approach breaks down as the reflection points are no longer at the source-receiver midpoint, and moreover the reflection point varies with source-receiver offset, an effect known as reflection-point smearing. Despite these effects, a best-fit hyperbola can still be calculated that maximises the summation energy along its trajectory. This hyperbola is parameterised by its intersection with the zero-offset axis and a parameter with the dimensions of velocity, which is referred to as the stacking velocity.

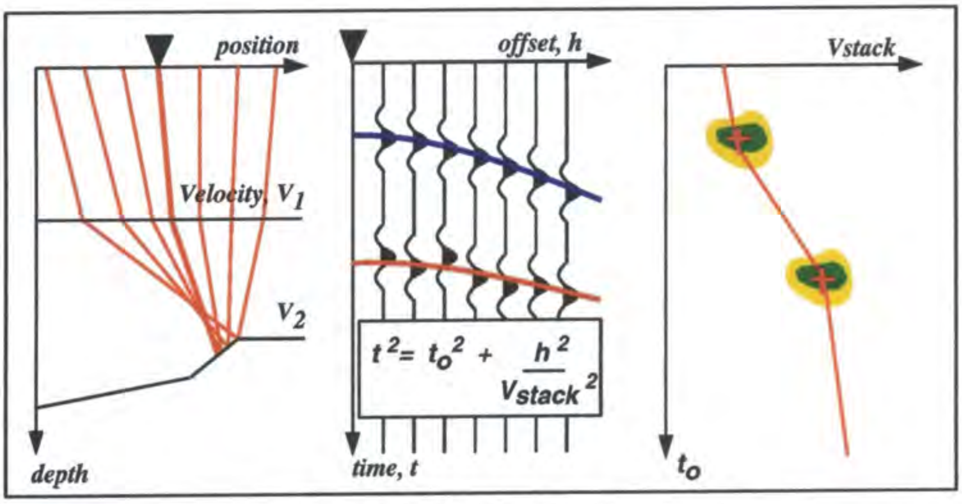


*Figure 2-3: Extension to real earth*

It is important to realise that, in general, the stacking velocity is not simply related to the propagation velocity of the earth, although in conventional Dix-based inversion schemes (Dix, 1955) it is often assumed to be equal to the vertical RMS velocity. In some sense it depends on the velocities within the offset ray fan, but it also contains structural terms. The stacking velocity is simply the processing parameter that maximizes the summation energy along a hyperbolic trajectory on the CMP gather, although in complex cases there remains a fair degree of subjectivity about where to pick, and this manifests itself as considerable scatter.

A stacking velocity field can be generated in 3D by defining a set of stacking velocity picks at a number of velocity analysis (VA) locations that specify the best hyperbolic approximation to the traveltime-offset curves for the main reflectors on the CMP gathers (see for example figure 3-1).

This field is subsequently used to perform normal moveout (NMO) corrections, which attempt to remove the offset dependence of the traveltime-offset curve by "flattening" the reflectors on the gather, prior to stacking together (averaging) all the traces with the same midpoint to produce a stacked section. The NMO is invariably to zero-offset, and the stacked-section is often loosely referred to as a zero-offset section. For the purposes of the inversion presented here, the two are assumed to be equivalent. This stacking procedure is primarily to improve the signal-to-noise ratio of the data and to reduce the storage size of the dataset making it more manageable and quicker to manipulate.

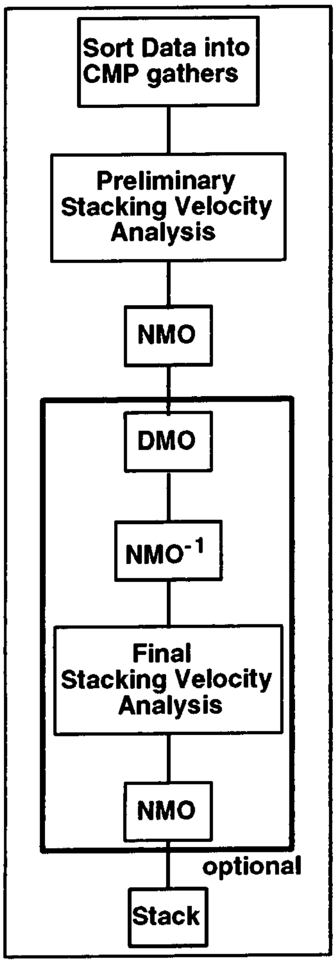


*Figure 2-4: Standard processing Flow*

The validity of the stack is of course dependent on the hyperbolicity of the traveltime-offset curve. Berkhout (1980) presents several examples of structure that cause the CMP principle to fail significantly and reflection-point smearing to be very large. In these cases, the redundancy in the data implicitly assumed by the stacking procedure is false and a great deal of valuable information is lost.

The reflection-point smearing and dip-dependency of the stacking velocity field were studied by Judson et al. (1978) and led to the, now standard, processing procedure known as dip moveout (DMO) (Deregowski and Rocca, 1981). DMO is a rebinning process that improves the quality of the stack by attempting to map common-offset sections into equivalent zero-offset sections (Deregowski, 1986). The DMO correction operates under many simplifying assumptions about the velocity model and does not usually remove all the structural effects from the stacking velocities (see Section 3.2.2). Nevertheless, the CMP principle is generally a much better approximation after DMO corrections have been applied.

Although it has long been appreciated that the traveltime-offset curves are significantly non-hyperbolic, standard NMO analysis continues to be used. Alternatives to the standard NMO equation have been proposed, such as the three-parameter equation (Castle, 1988) which has been demonstrated to improve the stack quality and to make the estimation of the stacking parameters more reliable (Thore et al., 1994). The continued success of NMO relies on the fact that it works surprisingly well even in fairly complex structures with strong lateral variation. In recent years, however, as the maximum source-receiver offsets have increased, the effect of the non-hyperbolicity becomes more significant.

Subsurface structure causes the reflection point of an event to be very different from the midpoint of the source and receiver that measured it, so the stack section is not a good representation of the subsurface geometry and consequently is notoriously difficult to interpret. The existence of triplications makes picking extremely complicated, especially in 3D. The stack domain also contains considerable noise from diffraction events so the lateral resolution is poor, making short wavelength events impossible to pick with confidence (Whitcombe, 1994).

### 2.2.2 The Migrated Domains

Migration processes attempt to move data from their midpoint (or stack) positions to their correct positions in the subsurface. Migration depends strongly upon a velocity model of the subsurface; however, in complex areas where migration is most needed, the velocity field may be poorly determined. Fortunately, the process of migration generally helps to clarify structure and makes interpretation much simpler even when the velocity model is incorrect, although the events will be mispositioned. In other words, migrated sections, even when sub-optimal, provide a much more realistic picture of the subsurface than the unmigrated stacked section, and correspondingly are far easier to interpret.

There are two classes of migration operator: depth-migration and time-migration. The difference between them arises from the way in which they handle lateral variations in the migration velocity model. Depth-migration, the more accurate technique, takes the velocity model fully into account, whereas time-migration approximates lateral variation in a variety of ways. Due to these approximations, time-migration is considerably cheaper to perform than depth-migration and therefore is by far the more commonly used migration technique, especially for 3D data. When significant lateral velocity variations exist, however, the time-migrated events will be mispositioned. The magnitude of this mispositioning and what constitutes significant lateral variation will be discussed later in this chapter. Time-migration continues to be widely used because it eases geological interpretation without requiring a very sophisticated velocity model. In fact, because time-migration is less sensitive to errors in the velocity field than depth-migration, the time-migrated section is often more interpretable than a sub-optimal depth-migration in spite of the stronger approximations.

## 2.3 Mapping between the Stacked and Depth-Migrated Domains

When a seismic reflection event is recorded at the surface, the time delay between adjacent receivers can be used to determine the direction of the best estimate of a local plane wave incident at the observation point (figure 2-5). From the geometry of the figure, the traveltime gradient with respect to the surface position is simply related to the angle of propagation of the plane wave by

$$\frac{dt}{dx} = \frac{\sin\theta}{V} \tag{2-1}$$

where $V$ is the local propagation velocity. The reciprocal of this traveltime gradient is the speed with which the event appears to travel along the surface and is referred to as the apparent velocity.

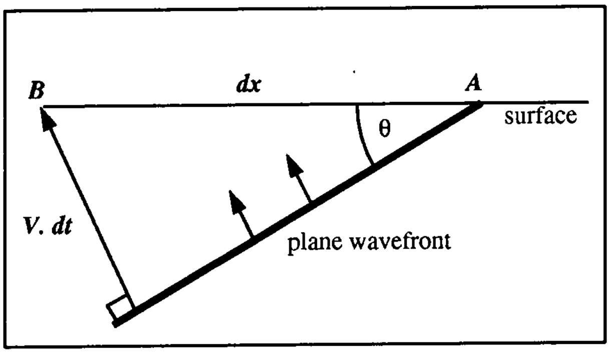


***Figure 2-5:** Calculation of initial direction of propagation from traveltime gradient: the plane wavefront hits the surface point A at time dt before it reaches point B, a distance dx away.*

The process of depth-migration attempts to back-propagate the recorded event to find the true position of the reflection point. If we consider a zero-offset event, i.e. coincident source and receiver positions, then the time taken for the energy to travel from the reflection-point to the receiver (the one-way time) is simply half the source to receiver traveltime. Given a velocity model of the subsurface and the traveltime gradient, which from equation 2-1 gives the initial angle of propagation, it is possible to trace a ray down from the surface, refracting in accordance with Snell's law, until the one-way traveltime is exhausted.

The end point of this zero-offset ray represents a point on the reflector, and the envelope of many of these points represents the topology of the reflector. In isotropic media, zero-offset rays meet the reflector at right-angles (the angle of incidence equals the angle of reflection), and are referred to as normal rays. Each zero-offset ray also gives the local dip of the reflector at the end of the ray. This dip information could be used to help constrain the reflector geometry. In practice, however, such an approach is expensive and it is usually more efficient to fire a large number of closely sampled rays, which should be sufficient to ensure the dip of the layers in depth is estimated to the required accuracy. This process is commonly referred to as map migration.

If the stacked section is assumed to be equivalent to a zero-offset section, then interpreted events in the stacked section can be used as the input to map migration. In conclusion, this poses a dilemma: to accurately back-propagate post-stack events into depth we need interpreted data in the stack domain. However, for the reasons mentioned earlier, obtaining such interpreted data is difficult and time-consuming making it undesirable in practice.

## 2.4 Mapping from the Time-Migrated to the Depth-Migrated Domain

To avoid the difficulties of picking in the stack domain, interpretation has traditionally been carried out in the time-migrated domain. The stacked section is deconvolved and time-migrated using a velocity field that is generally based upon a smoothed version of the stacking velocities, give or take, as noted by Hatton et al., (1986): "an apparently random few percent to account for the day of the week, the weather, or some other whim". The strongest reflectors on this migrated data block are picked, and define horizons that describe the major acoustic impedance contrasts. The question then is: how can the interpreted input in the time-migrated domain be mapped into depth?

If the process of time-migration works perfectly, then all of the energy from a diffraction curve is placed at its apex where the time-dip, by definition, is zero. After time-migration, therefore, the apparent velocity of the event becomes infinite, and with reference to equation 2-1 it is clear that a perfectly time-migrated section can be depth-migrated by firing rays vertically downwards from the surface. This type of ray is called an image-ray (Hubral, 1977). Image-rays should be traced, refracting through the velocity model, in exactly the same way as the other zero-offset rays discussed previously. The image rays are adding back some of the refraction effect that is partially neglected by time-migration.

If there is no lateral variation in the velocity model, the image rays stay vertical and the depth section is simply related to the migrated-time section by vertical stretching (i.e. the time-migration has positioned the events correctly). It is usual industrial practice to produce the depth maps by vertically stretching the time-migrated maps, implicitly making the assumption that lateral variations in the migration velocity model are negligible. This is only suitable for gently dipping structures with very long wavelength velocity variations. The difference between the vertical stretch and the image ray is known as the image ray correction.

The use of image rays assumes that the time-migration operator worked perfectly. In cases where lateral velocity variations exist, however, time-migration cannot work perfectly even if the velocity model is known exactly. In practice, the migration velocity model is far from optimal and the combination of these effects strongly limits the applicability of image rays. In the next section, therefore, I consider more accurate techniques that try to simulate more faithfully the route followed by the actual seismic data.

## 2.5 Mapping between the Time-Migrated and the Stacked Domains

Direct mapping from the migrated-time domain to the depth domain makes strong and commonly invalid assumptions about the time-migration procedure. The only accurate way to perform this mapping is to first transform the time-migrated interpretations back into the stacked domain before using normal rays to go into depth. This section explains the first stage in this procedure: the link between the stacked and time-migrated domains.

It is important to realise that the approximations made during time-migration depend upon the actual migration algorithm being used. Consequently, it is not possible to produce a simulation technique that is exactly valid for all types of time-migration. In particular, the Kirchhoff and finite-difference operators are shown to give significantly different results in cases where strong lateral velocity variations exist.

### 2.5.1 Kirchhoff-Type Time Migration

In the case where velocity varies only with depth (i.e. there is no lateral velocity variation), the traveltime curve for a diffracting point depends only upon the velocity field above the diffracting point. This diffraction curve represents the zero-offset response of the point. A perfect migration algorithm would sum the amplitudes along the curve and place the result at the apex (Hagedøørn, 1954). If the apex is not an actual diffracting point, the values along the curve will not be systematic, i.e. there will be no coherency along the curve, and positive and negative values will tend to cancel.

Huygens' principle means that a reflector can be thought of as a set of closely spaced diffracting points and a reflection as the interference composite of their diffraction responses. Thus if reflection events are migrated as if they were diffraction events then they will be migrated correctly. This is the basis of diffraction-stack migration schemes which simply scan over the stack section summing along diffraction curves defined by the migration velocity field. These summation techniques, based on ray theory, were refined by Kirchhoff integral theory which made the summation consistent with the wave equation by including the effects of spherical spreading and obliquity. These refinements are not important for the kinematic discussion presented here.

A processing operator, such as Kirchhoff migration, can be thought of in terms of its impulse response. This response defines the action of the operator on a single sample of data (an impulse). The processing procedure often works by application of this response to every sample in the seismic domain, and summing the results. Due to destructive interference, only a small area of the operator's response contributes significantly to the final image. The significance of this is that the areas of constructive interference can be identified directly if events are characterised with a dip as well as a position. In other words, the region of coherency is positioned where the impulse responses of two points, infinitesimally separated along the dip direction, become tangential.

Throughout the inversion we are dealing with interpreted data and consequently these dips can be readily calculated. A recurring technique presented in this thesis is to work out the kinematic response of the different processing operators as a function of the dip and position of the event. This is an important concept and therefore we talk about the simulation of the processing operators rather than their straightforward application. This again restricts the discussion to kinematics only. In Kirchhoff-type time migration it is possible to analytically calculate how a given time-dip migrates with a given migration velocity (figure 2-6). The hyperbolic summation trajectory of the Kirchhoff operator, shown on the bottom panel, can be written as

$$t_s^2 = t_m^2 + \frac{4(x_s - x_m)^2}{V_m^2}, \tag{2-2}$$

which relates the unmigrated position, $(x_s, t_s)$ to the migrated position, $(x_m, t_m)$ by the migration velocity, $V_m$. The figure also shows how a given time-dip is associated with a unique raypath.

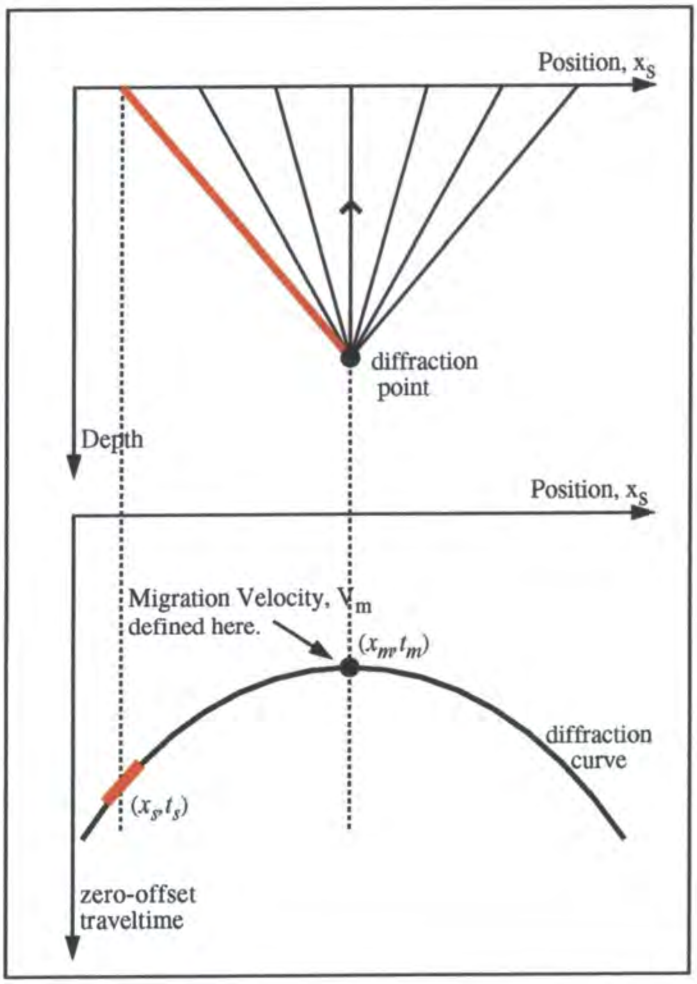


*Figure 2-6: Schematic showing how an event with a given time-dip is migrated by the Kirchhoff operator*

Differentiating equation 2-2 with respect to $x_s$ gives the relationships

$$x_m = x_s - t_s \beta_s \frac{V_m^2}{4} \tag{2-3}$$

and

$$t_m = t_s (1 - V_m^2 {\beta_s}^2 / 4)^{1/2} , \tag{2-4}$$

which give the migrated position and time as a function of the stack position, stack time and the stack time dip, $\beta_s = \frac{\partial t_s}{\partial x}$. These equations can be used to model how events move when time-migrated. There is an added complication in that the migration velocity, $V_m$, is defined at the apex of the migration hyperbola, the position of which depends upon the migration velocity used. This means that the solution must be iterative unless the migration velocity is constant. The time-migration velocity model is generally quite heavily smoothed and so an iterative approach is very stable and converges rapidly.

A similar approach can be used to back out the effects of time-migration, a process known as demigration. Whitcombe (1994) reversed equation 2-3 and equation 2-4 to give

$$x_s = x_m + t_m \beta_m \frac{V_m^2}{4} \tag{2-5}$$

$$t_s = t_m (1 + V_m^2 {\beta_m}^2 / 4)^{1/2} . \tag{2-6}$$

These allow an event defined by its time, $t_m$, and time-dip, $\beta_m = \frac{\partial t_m}{\partial x_m}$, to be mapped from migrated-time back into the zero-offset domain. The new time dip is conveniently given by

$$\frac{\partial t_s}{\partial x_s} = \frac{t_m \partial t_m}{t_s \partial x_m}, \qquad (2\text{-}7)$$

which relates equation 2-3 and equation 2-5.

In this case the migration velocity to be used in the mapping is known *a priori* and the solution is analytic. This demigration technique is very important as it makes it possible to take the real sub-optimal migration velocity field into account and not make assumptions based on an idealised view of time-migration. Whitcombe's (1994) demigration approach is only strictly valid for a locally constant velocity field. It is easily extended to allow for variable fields which requires the addition of a second order term involving the migration velocity gradient. This term is generally small and is only significant when the time-dip is high and the migration velocity field is changing rapidly.

The above analysis is presented in 2D for simplicity, but it is easily extended into 3D as the migration response is effectively a 2D operator that works in the plane of maximal time-dip. In practice, the migration velocity field is usually generated as a heavily smoothed version of the stacking velocity field on a 3D grid. However, as the migration velocity is defined at the migrated position, it is far more convenient to provide maps of the migration velocity field sampled on to the migrated time-interpretations. This interpolation and resampling is done prior to inversion.

Whitcombe (1994) noted that although these equations only apply to a Kirchhoff time-migration, they could be applied successfully to migrated sections produced by a finite-difference algorithm, so long as the migration velocity field remains smooth and the time-dips are not too large. In order to check this claim, I have compared this approach to more complex techniques that try to specifically simulate the kinematics of the finite-difference operators.

### 2.5.2 Finite-Difference Time-Migration

Finite-difference solutions to the scalar wave equation based on wavefield extrapolation have been pioneered by Claerbout (1970,1971). They are known from both theory and practice to give different results from Kirchhoff migration due to the different approximations made during the handling of lateral variations in the velocity model. For this reason, techniques have been developed that model the kinematics of finite-difference operators. The simulation can be done by anisotropic ray tracing either in depth (Khare, 1991) or directly in the time-migrated domain (Raynaud and Thore, 1993).

Finite-difference operators work by a successive redatuming of the seismic data. Consider a zero-offset seismic section. The vertical axis is the traveltime and the horizontal axis is the source/receiver position. Through use of the wave equation and a knowledge of the velocity model, this seismic section can be extrapolated downwards to produce the section that would have been recorded if the sources and receivers had been placed, not at the surface as in the physical experiment, but at some finite distance, $\Delta z$, beneath it. Then, just as events at zero time on the original section are by definition at zero depth, events on the redatumed section at zero time will be at a depth of $\Delta z$. This is called the imaging principle.

Once the redatumed data volume has been calculated the procedure can be applied recursively to produce sections at $2\Delta z, 3\Delta z$ and so on. For each of these sections, the imaging principle is applied and the seismic data at zero time associated with the redatuming depth. In this way the seismic data can be depth-migrated. The commonest approach assumes that the velocity is constant within any given $\Delta z$ interval. The step-length is chosen so as to ensure that this is a good approximation. For this constant velocity case, the downward-extrapolation operator is simply a hyperbola. To explain the simulation of the finite-difference time-migration operator I would first like to use the analogy of simulating finite-difference depth-migration by ray-tracing.

### 2.5.3 The Link between Ray-Tracing and Finite-Difference Depth Migration

The concept of using ray-tracing to perform depth-migration is well known and understood. Figure 2-7 shows the link between ray-tracing and finite-difference depth-migration. The top and bottom panels of the figure show the zero-offset and depth domains, respectively. Consider an event in the zero-offset domain characterised by its time and time-dip. This event can be back-propagated through a layer of thickness $\Delta Z$ with a constant velocity $V_1$. If the propagation direction is vertically downwards then this takes a time, $t_1 = \Delta Z / V_1$. For off-vertical propagation directions the time taken is longer and the event maps to a hyperbola which is a function of this direction. When we know the time-dip of the event, we know its propagation direction and the mapping is uniquely defined. This mapping, when viewed in the depth domain, can be thought of as a ray in the sense that it is the trajectory along which energy propagates during the downward continuation of the finite-difference process. The procedure is applied recursively until the traveltime of the event is exhausted and the imaging criterion satisfied.

At each stage of the process the time-dip must be recalculated due to refraction of the wavefront. When considering how the finite-difference migration will operate on an event with a given time-dip, we must again consider how two operator responses infinitesimally separated along the time-dip direction will constructively interfere. In a horizontally stratified model, without lateral velocity variations, the shape of the finite-difference operator remains the same within any given $\Delta Z$ interval and, therefore, the operators constructively interfere in such a way to keep the time-dip constant throughout the back propagation of the event. When lateral variations exist in the velocity model, the shape of the finite difference operator changes as a function of lateral position and the constructive interference can occur at a different time-dip. This behaviour, for a given event, is exactly analogous to ray-tracing through the velocity model and applying Snell's laws at velocity discontinuities to allow for the refraction. The kinematics of finite-difference depth-migration, therefore, can be simulated by ray-tracing.

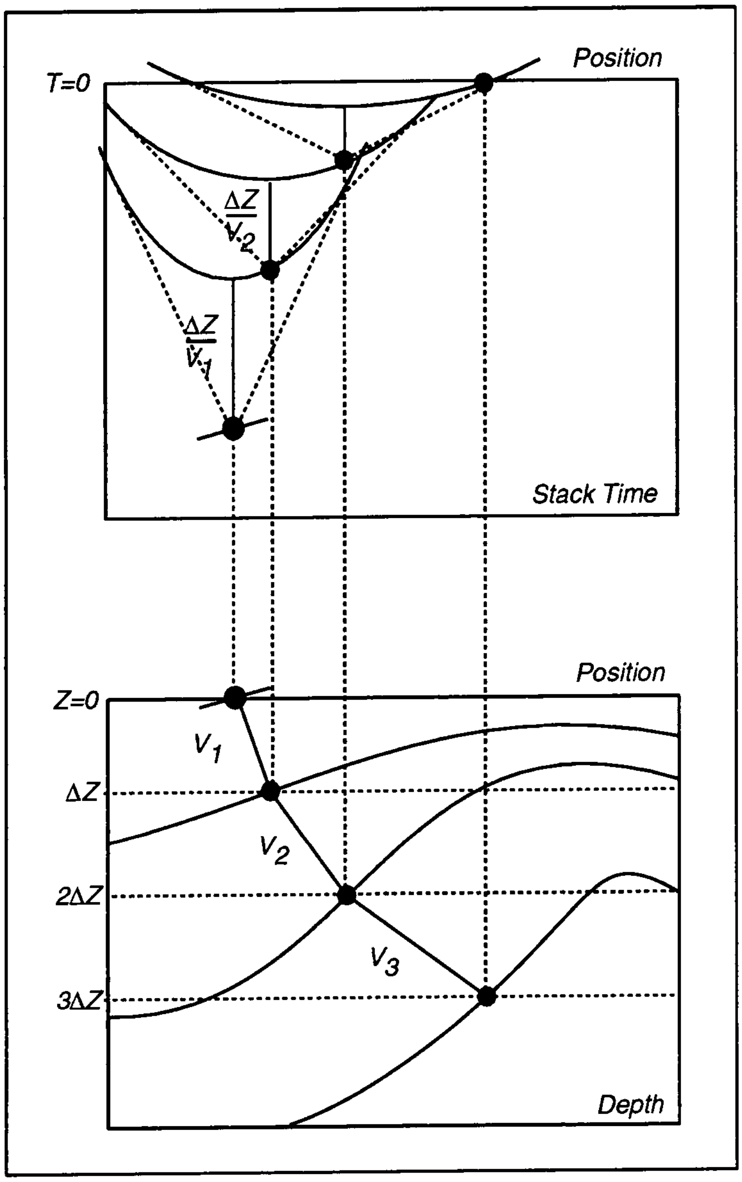


*Figure 2-7: Simulating finite-difference depth-migration by ray-tracing*

### 2.5.4 The Link between Ray-Tracing and Finite-Difference Time Migration

Similarly, the finite-difference time-migration operator can be simulated by ray-tracing. The principle is shown in figure 2-8, and helps to give us a real physical insight into the difference between time- and depth-migration. In time-migration, the back propagation is through a layer with a thickness defined in vertical time, not depth. It is this lack of physicality that results in lateral variations not being correctly handled.

As with depth-migration the back-propagation can be represented by a ray. This ray, however, is travelling in the time-migrated domain, and as such is not physically meaningful as the dimensions of the vertical and horizontal axis are different. This dimensional mismatch effectively introduces a vertical scaling factor that means that the ray can be modelled as if it were travelling in an elliptically anisotropic medium.

To explain this, consider figure 2-9. The local impulse response of a perfect time-migration (valid for all dips) is an ellipse with a horizontal axis of length equal to the local migration velocity and a vertical axis of length equal to unity. A given time-dip in the zero-offset domain maps to a single point on this impulse response. Again, the curve joining the stack and migrated positions can be thought of as a ray. This ray is in the time-migrated domain where the horizontal propagation velocity is the time-migration velocity and the vertical velocity is unity, since the traveltime of a vertical ray is unchanged by the time-migrated process. The finite-difference time-migration operator can be simulated in a similar way to the depth-migration operator but the ray-tracing must be done in the time-migrated domain through an elliptically anisotropic velocity model.

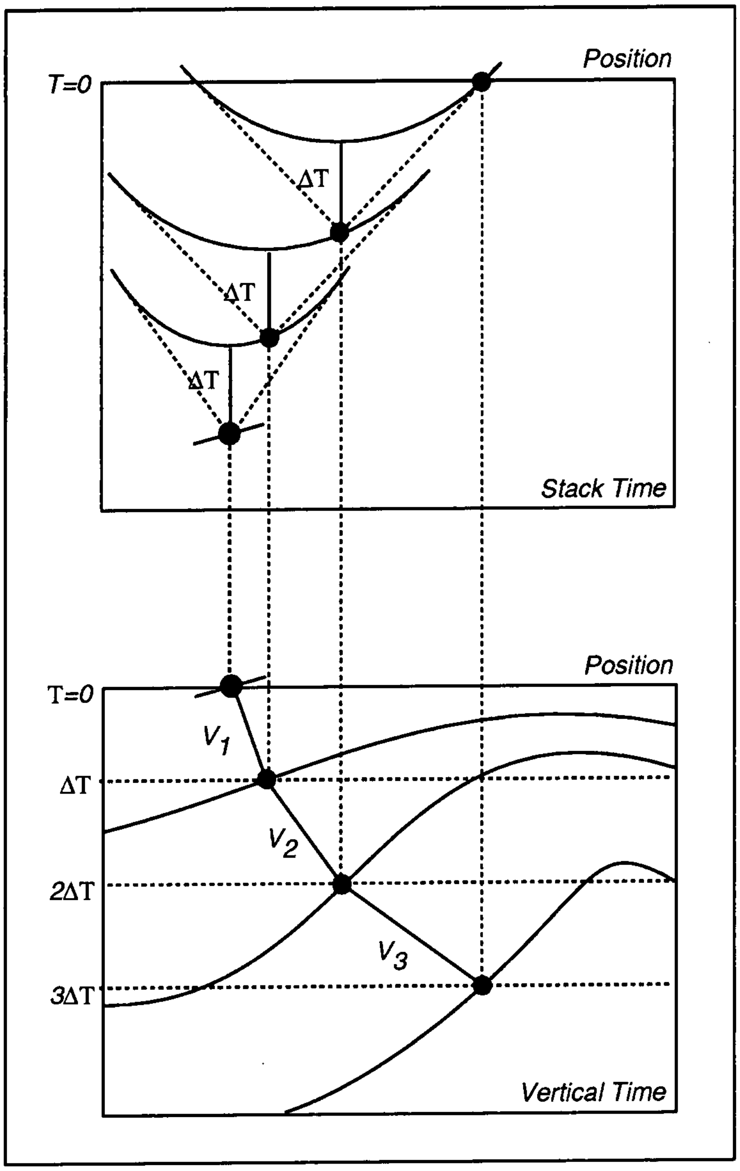


*Figure 2-8: Simulating finite-difference time-migration by ray-tracing*

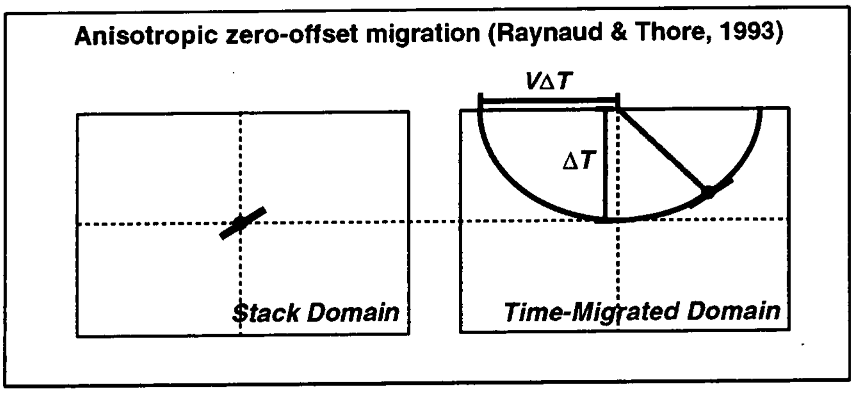


*Figure 2-9: Impulse response of a perfect time-migration operator*

As a first approximation, the local impulse response of the time-migration operator can be assumed to be elliptical, the case for a perfect time-migration algorithm. If more accuracy is needed, the actual impulse response of the finite-difference scheme can be used instead. Such a refinement is possible but could become computationally cumbersome, and anyway the non-ellipticity will only become important for high dips and very severe lateral variations which are, in general, beyond the practical limits of time-migration (Raynaud and Thore, 1993).

The difference between time- and depth-migration is often explained by saying that time-migration simply ignores Snell's law. The above analysis shows that this is not correct in the case of the finite-difference operator. Instead, there is still a form of refraction effect, but the time-migration follows a modified Snell's law different from that of depth-migration (Khare, 1991). This simulation technique is very powerful in that it can simulate the time-migration or demigration of any approximation to the scalar wave equation at a much lower cost.

### 2.5.5 Comparison of Kirchhoff and Finite-Difference Simulation on a Synthetic Example

To illustrate the difference between the Kirchhoff and finite-difference time-migration simulation techniques, I considered an example of a diffraction point in a simple two-layer model. A synthetic diffraction curve was computed by ray-tracing from the diffraction point. The two layers have constant velocities of 2000 m/s and 3000 m/s and are separated by a dipping layer at 27 degrees (figure 2-10), which provides a strong lateral velocity variation in the model. Figure 2-11 shows how finite difference time-migration can be simulated by ray-tracing through an elliptically anisotropic velocity model directly in the time-migrated domain.

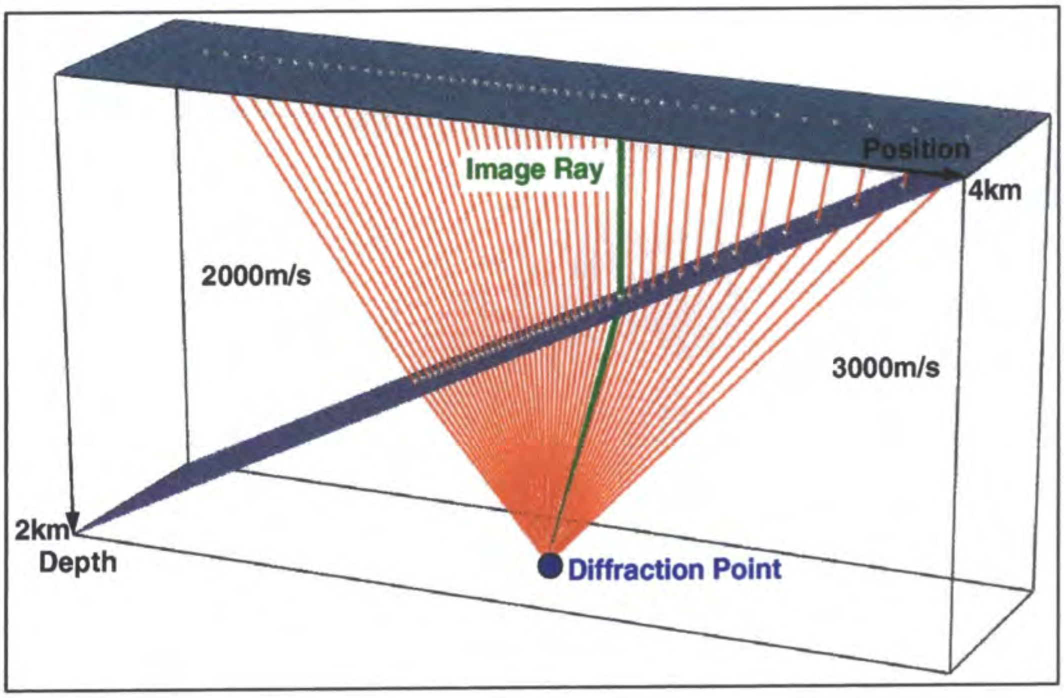


*Figure 2-10: Modelling of a diffraction point through a synthetic two-layer model*

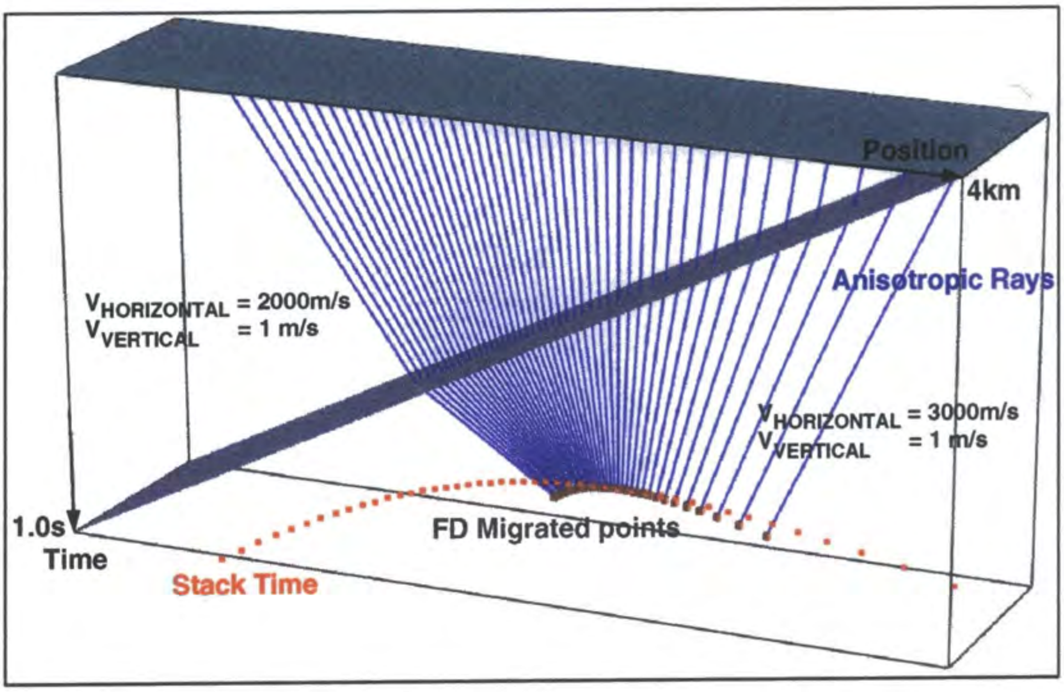


*Figure 2-11: Simulation of FD time-migration by anisotropic ray-tracing*

To check the accuracy of the simulation, the diffraction curve was convolved with a Ricker wavelet to produce a synthetic stacked section, and this stacked section was then time migrated using both Kirchhoff and finite difference algorithms using a commercial

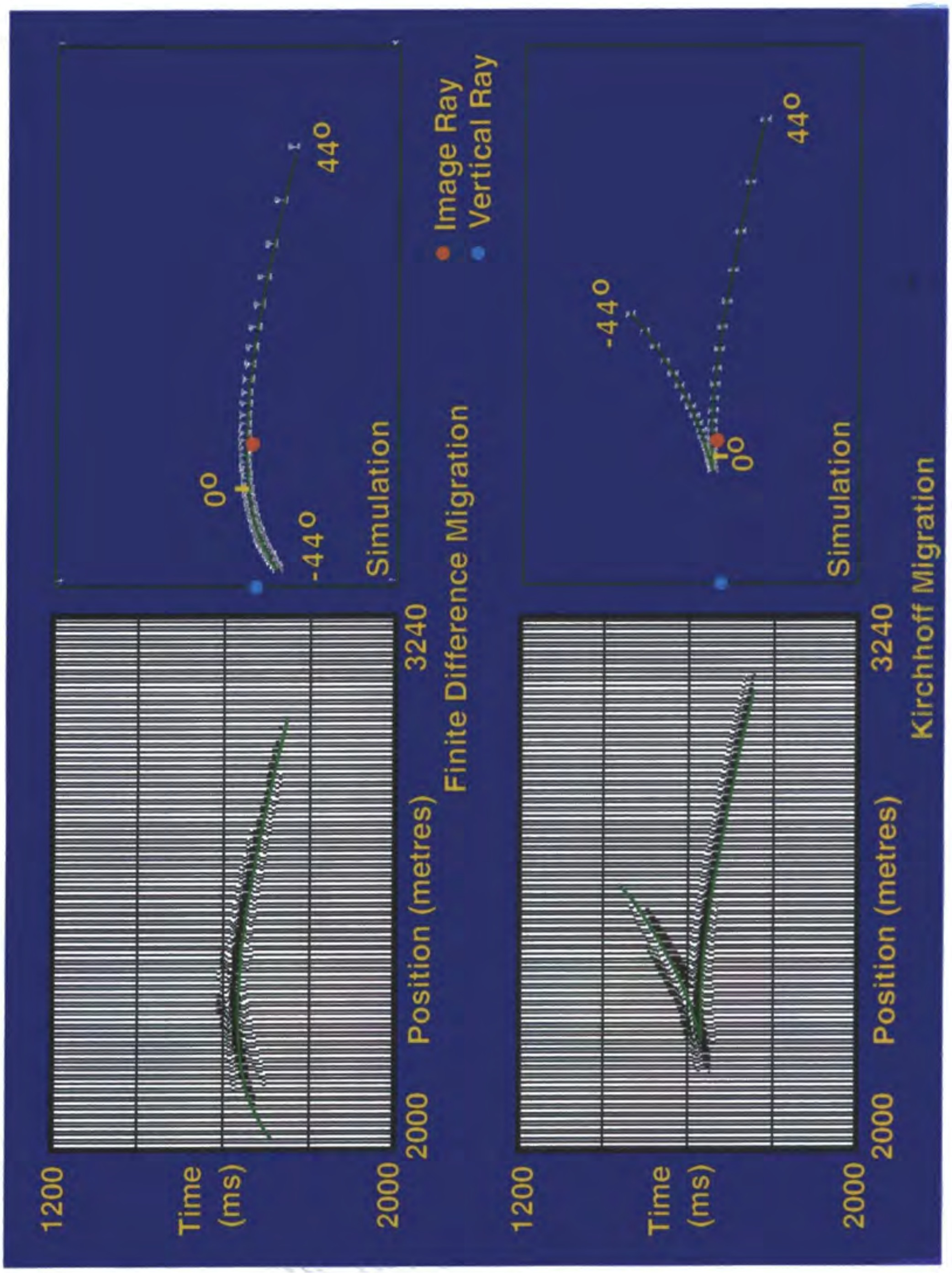


*Figure 2-12: Comparison of the simulated and actual time-migration responses for the two-layer synthetic model*

2D processing package, ProMAX$^{TM}$. The results, together with the simulated responses are shown in figure 2-12.

As expected the time-migration does not correctly refocus the diffraction point but instead produces a so-called 'plume effect' (Bevc et al., 1995). This demonstrates that the discrepancy between depth- and time-migration is dip dependent. Also, the migrated results are significantly different depending on which algorithm is used, demonstrating the importance of simulating the actual migration operator that was used in cases where strong lateral variations exist. The match between the migrations and their simulations is almost perfect, and calibrates the plume in terms of the real, geological dip. For comparison the results of the vertical stretching and the image ray simulations are also shown.

## 2.6 Application of Domain Mapping Techniques to the Inversion

As far as the inversion scheme is concerned, the major application of this work is to back out the effects of time-migration on interpreted events and thus obtain interpreted events in the stack domain suitable for subsequent ray-trace depth migration. The following sections describe the implementation of these techniques to the inversion problem and demonstrate their use on real data examples.

## 2.7 Demigration of Migrated-Time Interpretations

To explain the main geophysical processing carried out during the domain mapping, consider figure 2-13 which depicts the processing flow for a 2D synthetic syncline. The migrated-time interpretations are usually output from seismic interpretation packages as regular cartesian grids (top panel).

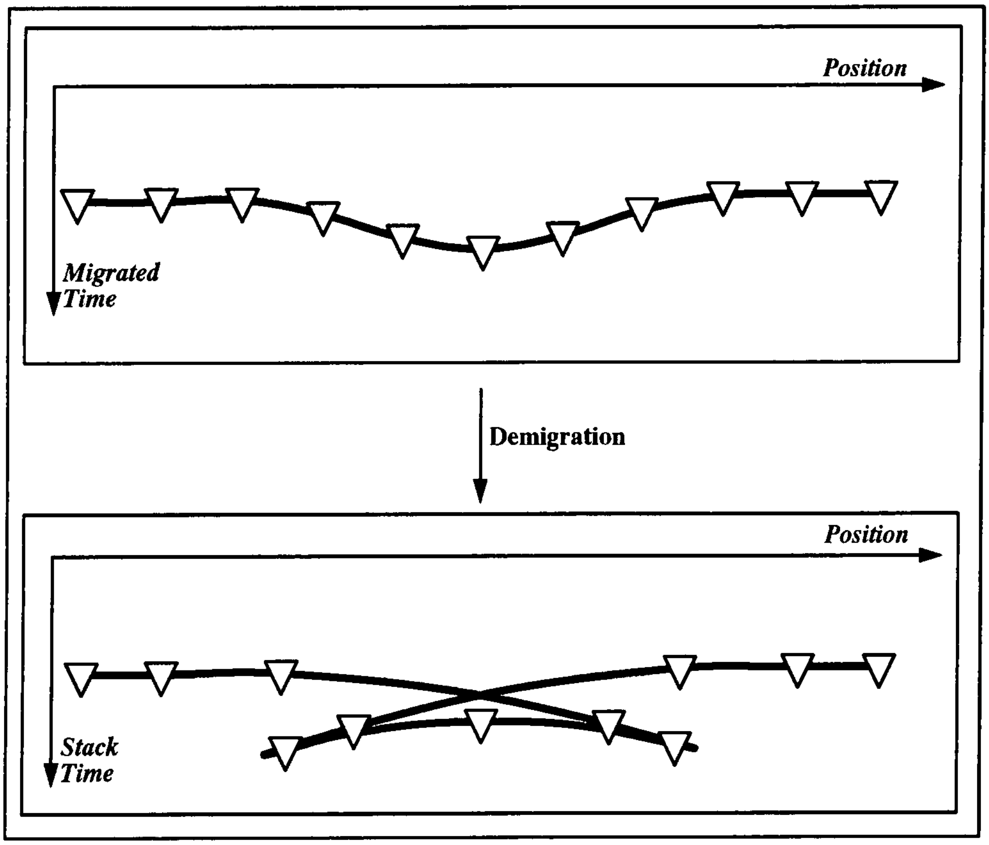


*Figure 2-13: Domain mapping from migrated-time into stack-time by demigration*

Using a horizon-consistent time-migration velocity field supplied as input, the time-migrated interpretation is mapped back into the stack domain, as described in Section 2.5. The demigration function associates with each point on the migrated-time grid a new position and time-dip in the stack domain, as shown in figure 2-13. The interconnectivity of the points in the stack domain is the same as that of the time-migrated domain. This one-to-one relationship between points in the seismic domains is a very useful construct and simplifies the subsequent modelling (see Section 3.2.1)

The figure shows the well known bow-tie response for the syncline. In the stack domain the horizons are generally multi-valued and not uniformly sampled in space. This simple example illustrates the added complexity of interpretation in the stack-domain. The demigration scheme circumvents these difficulties as the interpreter need not work directly in the stack domain. It is very important, however, that the demigration result be checked against the seismic data to ensure the quality of picking in the time-migrated domain. This stack domain is now treated as invariant during the inversion process.

Often, it is convenient to work on only small parts of the model or with a different grid sampling. This means that some interpolation must be carried out prior to the mapping. The calculation of the traveltime at a given location is done by means of a bilinear interpolation scheme (figure 2-14). Firstly, the grid cell that contains the point is found and then the value, $V$ is calculated from the values of the four surrounding grid nodes (figure 2-14):

$$V = \left(1 - \frac{y}{\Delta y}\right)\left[\left(1 - \frac{x}{\Delta x}\right)V_1 + \frac{x}{\Delta x}V_2\right] + \frac{y}{\Delta y}\left[\left(1 - \frac{x}{\Delta x}\right)V_3 + \frac{x}{\Delta x}V_4\right]. \tag{2-8}$$

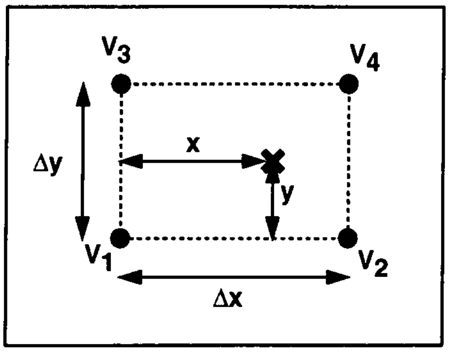


*Figure 2-14: Bilinear interpolation within the grid cells*

This allows the user to define a grid anywhere in the model with any sampling. The sampling can also be changed between different layers and between different areas of the same layer. This allows a coarser sampling to be used in the smoother areas of the model.

As well as the traveltimes, the demigration procedure requires the time-dips which are crucial for the mapping. These are calculated in a two-step process. Firstly, the gradients are defined at the grid-nodes using central differencing (top panel of figure 2-15): the dip at a node is the ratio of the time-difference between the previous and next nodes to twice the grid spacing. This approach can be modified to a one-sided calculation if both adjacent grid cells are not defined. Secondly, the variation of the time-dips within the grid cells is again calculated by bilinear interpolation. Curvatures and higher order derivatives can be found by recursively applying the same algorithms.

This simple interpolation method is very rapid and ensures derivative continuity across grid cell boundaries. The disadvantage is that when there is curvature present, the time variation is not exactly consistent with the time-dips. This can cause instability in the ray-tracing if the grid-sampling is too coarse or if the interpretation maps are not smooth.

Raw interpretations, especially when generated by automatic picking schemes, contain many high-frequency artifacts which cannot justifiably be interpreted given the limited bandwidth of the seismic data. In order to ensure lateral continuity between adjacent grid cells, it is good practice to select the grid sampling and level of smoothing so as to ensure that the highest frequency is less that half the Nyquist.

### 2.7.1 Fault Handling

A sampled dataset contains complete information about all spectral components up to the Nyquist frequency and aliased information about any signal components at frequencies higher than the Nyquist. The central differencing approach used to calculate the traveltime derivatives is only appropriate when the surface is not changing too rapidly. Faults, however, are very high frequency in the horizontal spatial coordinates, much higher than the Nyquist frequency of the grid in most cases. Central differencing techniques smooth out the time-dips across the faults and consequently smear out the extent of each fault's influence. As accurate fault positioning is very important for subsequent interpretation of the maps, this smearing is an unacceptable artifact.

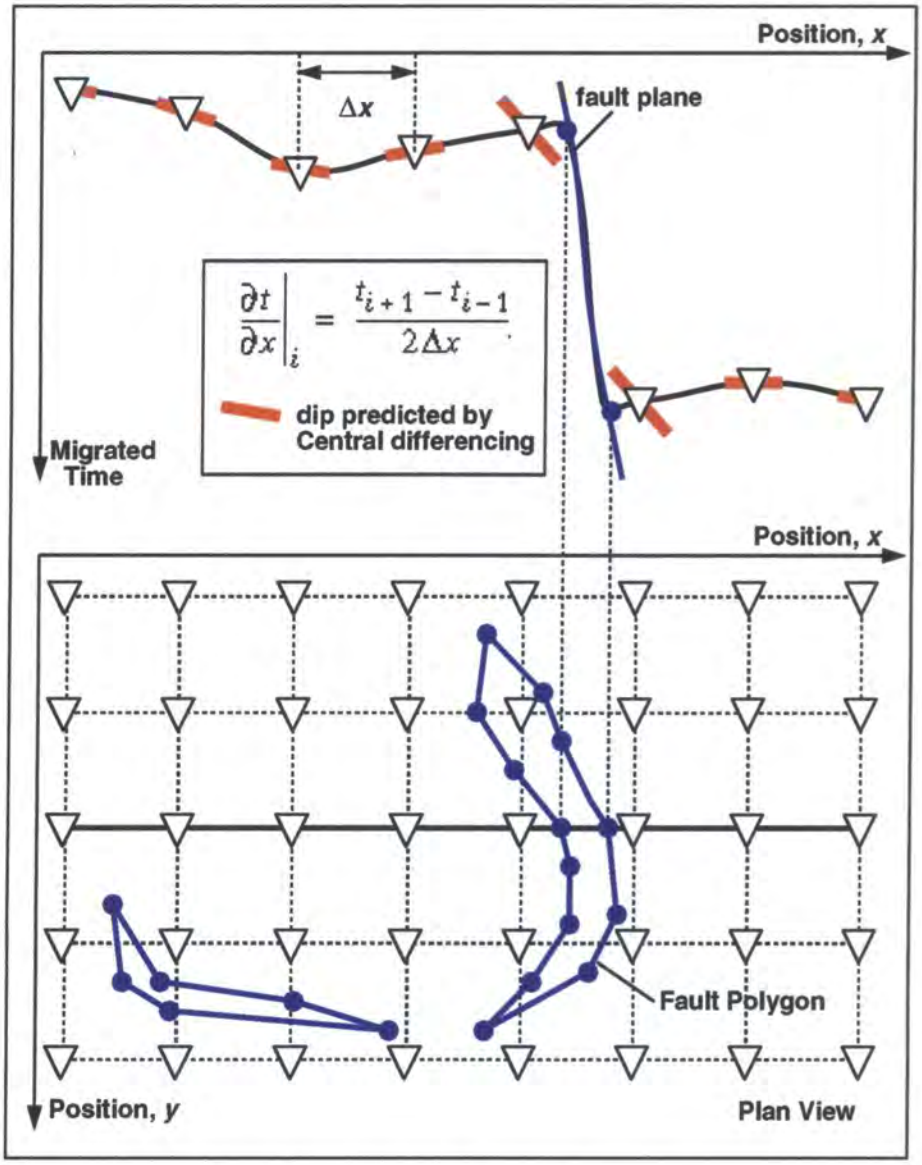


*Figure 2-15: Calculation of time-dips using central differencing.*

Figure 2-15 shows the problem at a fault lip. The time-dip estimated by central differencing at the grid cells adjacent to the fault is completely wrong and leads to significant mispositioning in the domain mapping. To prevent such effects, the positions of the faults can be specified accurately as polygons in the time-migrated domain. When calculating the time and time-dip at a given point, only points that are on the same side of a fault truncation are used. This approach is also applicable when the faults have a smaller lateral extent than the grid spacing. Points that are inside the fault polygon are generally removed.

This technique is a way of accurately reintroducing events that have higher frequencies than the grid's Nyquist frequency. The implementation of such a technique is very tricky numerically and great care must be taken to avoid problems with rounding errors.

### 2.7.2 Application to 3D Real Data Example

To illustrate the accuracy of the demigration procedure, figure 2-16 shows a real 3D data example from the North Sea. The top panel shows a cross-section of the migrated time block with the picked interpretations overlaid. This gives an idea of any possible ambiguity in the picking, and shows whether the fault planes are imaged or not.

Figure 2-17 shows the stack domain for the same 2D cross-section, with the demigrated response overlaid. The match is almost perfect, demonstrating the validity of the approach. In this case the data were time-migrated using a finite-difference algorithm and demigrated using the Kirchhoff operator. Contrary to the synthetic example presented in section 2.5.5, the lateral velocity variations are not too great in this example and the form of the time-migration simulation used is unimportant. In these cases the Kirchhoff approach is used as it is cheaper to perform.

Figure 2-18 shows a 3D time-migrated interpretation of around 100 $km^2$ from the same North Sea example. The structure is caused by salt uplift. The bottom panel shows the intersection of this surface with a plane running from west to east across the area. The interpretation shown was actually the second picked horizon: the intersection of the first with the plane is also shown as a red curve and is fairly flat over the area. Figure 2-19 shows the demigrated response of the interpretation shown in figure 2-18. This surface is fairly complex and there are many areas of triplication around the flanks of the salt. In this case, the grid sampling was chosen to be 50m x 50m which means that around 40,000 points were transformed. This takes considerably less than a second on a modern workstation. Again, the bottom panel shows the intersection of this demigrated surface with the same plane as in figure 2-18. This 2D cross-section is merely for the purposes of display since the demigration itself is in 3D. Two zones of triplication can readily be identified.

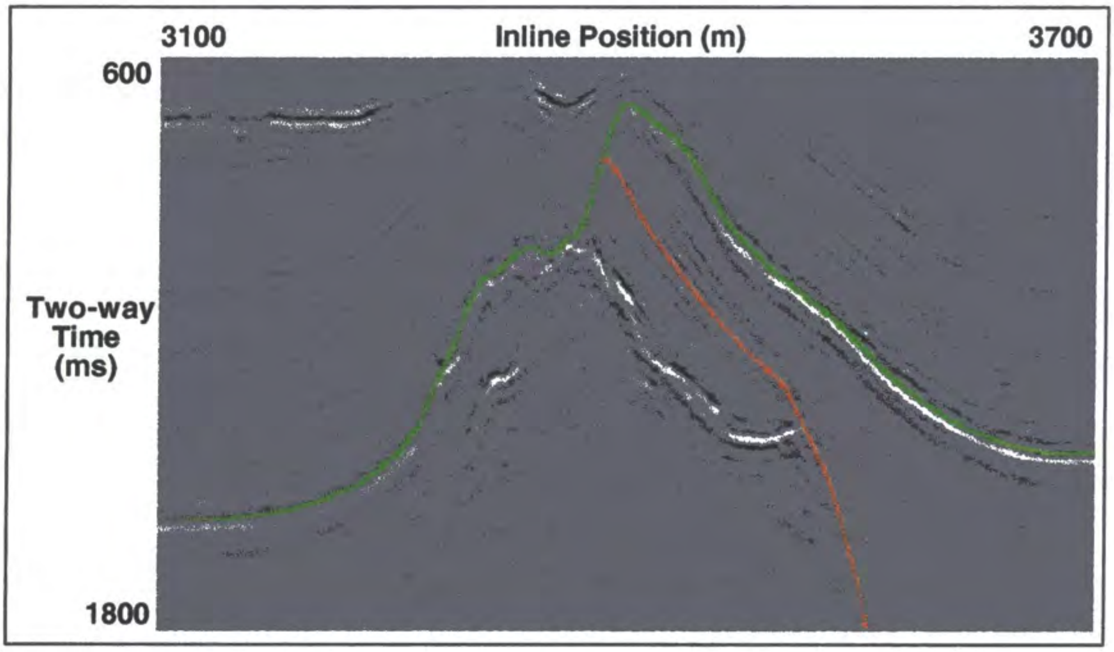


*Figure 2-16: Migrated-time section showing interpretation*

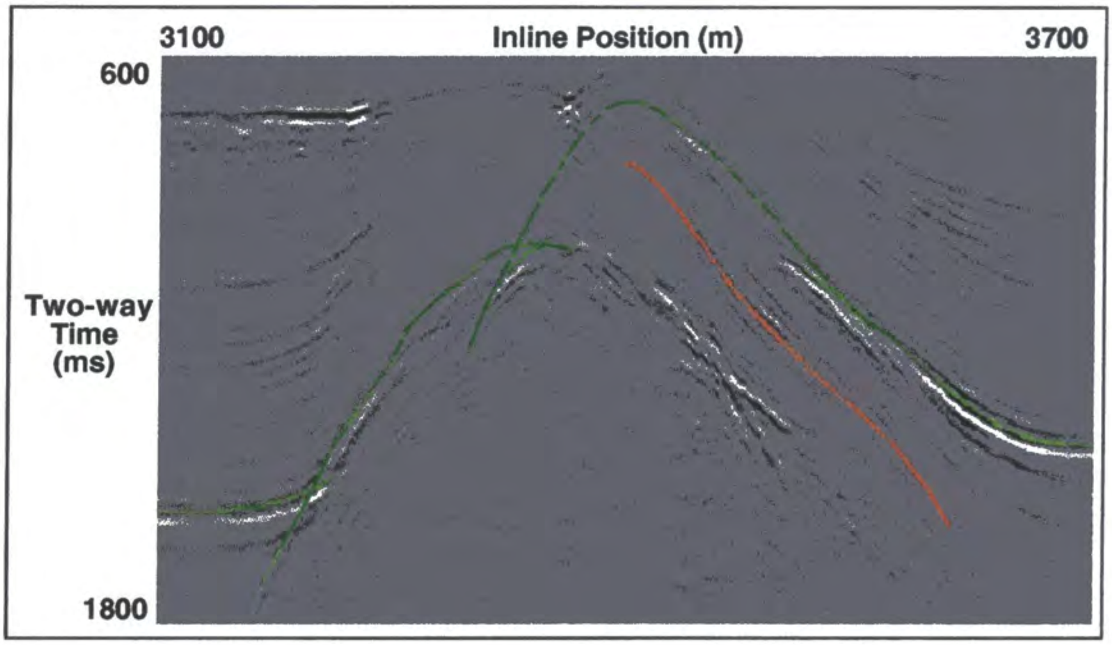


*Figure 2-17: Stack-time section showing demigrated response superimposed*

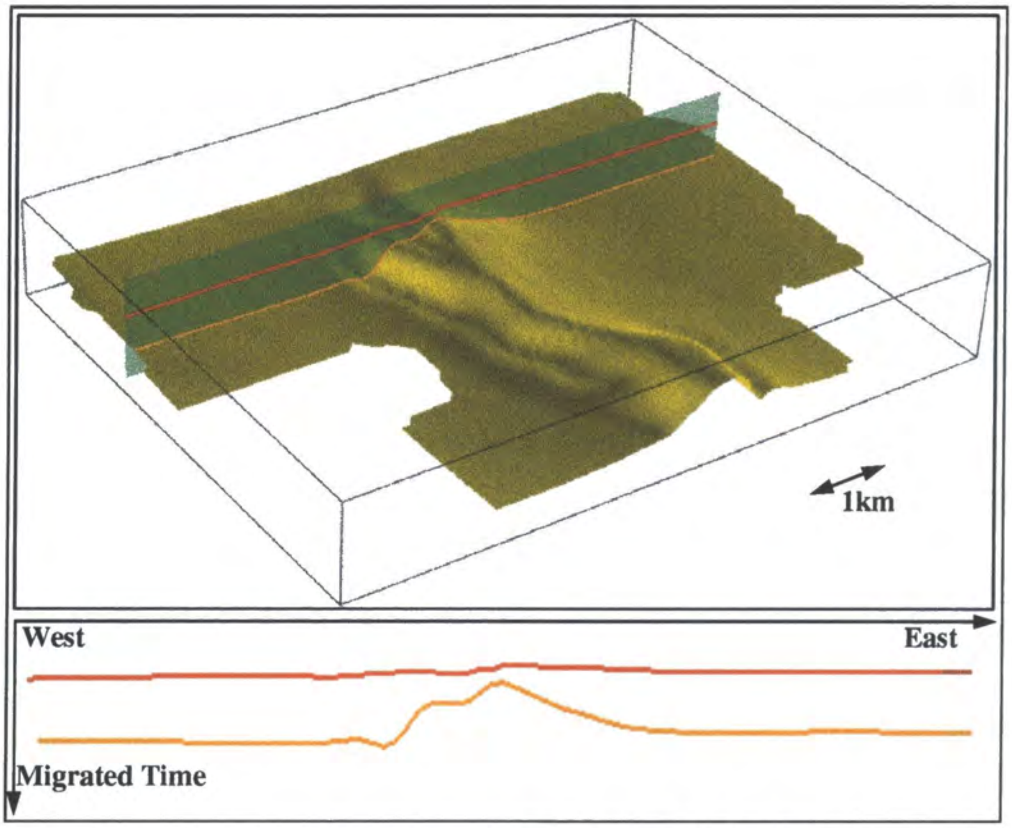


*Figure 2-18: Migrated-time interpretation from a real 3D North Sea dataset*

*Figure 2-19: The demigrated response of the interpretation*

## 2.8 Mapping of the Demigrated Stack Domain into Depth

Using an estimate of the interval velocity field, the horizon is mapped from the stack domain into the depth domain using normal rays. One normal ray is shot for each sample in the stack domain (figure 2-20). The initial angle of propagation of these normal rays is calculated from the time-dip associated with the point in the stack-domain by the demigration operator. In this way, each normal ray fired produces a single point in depth. These points will be referred to as crude depth points and their envelope represents the topology of the reflector. This shooting procedure is described in more detail in section 3.8.2.

The regularity of the points in the depth domain is indicative of the degree of equivalence between the time- and depth-migration. In the case of the synthetic syncline shown here, the migration velocity used in the demigration has been chosen to be equal to the constant depth-migration velocity. The crude depth points, therefore, move back to the same lateral positions they had in the time-migrated domain and the resultant depth surface is simply a vertical stretch of the time-migrated model.

Generally, when the time-migration velocity model is different from that of the depth migration or the process of time-migration does not work perfectly, the crude depth points in depth will be irregular. Again, we can take advantage of the connectivity relationships between the points and produce a depth surface by ‘joining the dots’. The rectangular grid cells of the migrated-time domain are mapped into quadrilaterals with the same connectivity relationships. If the effective depth-migration velocity differs significantly from the time-migration velocity model, then the final depth surface may still have gaps or residual triplications. From the connectivity relationships such zones can be identified and, if necessary, some interpretative input may be used to select which parts of the depth surface are considered reliable.

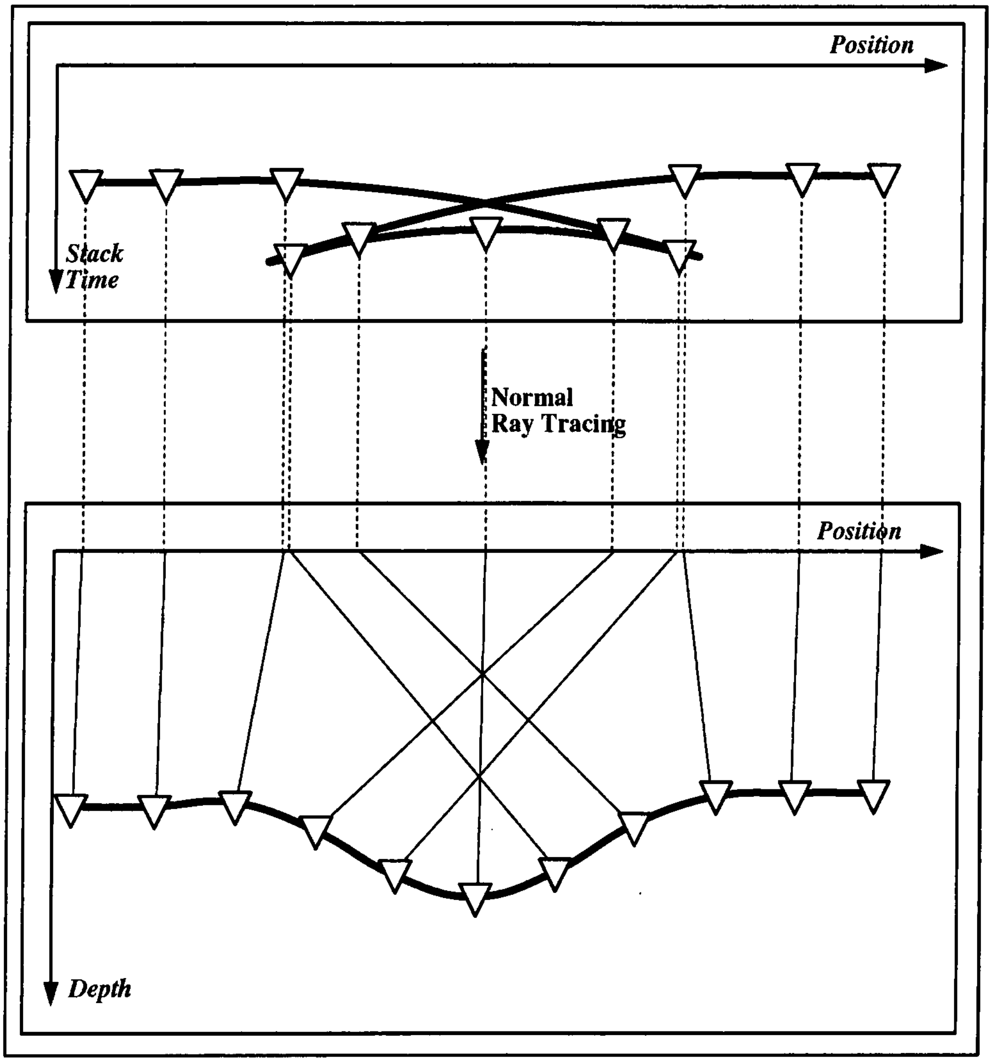


*Figure 2-20: Mapping of the stack-section into depth with normal rays*

This has a very important practical consequence when defining the time-migration velocity model. It is clear that when time-demigration followed by ray-trace migration is used, the migration velocity should be chosen so as to give the best image. Scaling of the migration velocity model so as to be consistent with the wells is a mistake, and can result in some data not being used by the inversion procedure.

The perturbation of the crude depth points away from a regular grid also gives a good indication of amplitude anomalies due to the inaccuracy of the time-migration process itself. Clustering together of the crude depth points in an area means that there is an amplitude deficiency on the time-migrated section and vice versa.

### 2.8.1 Resampling of the Depth Surface onto a Regular Grid

The subsequent processing of the depth section requires that the geometry is a single-valued function of lateral position. The geometry is therefore resampled on to a regular grid, the extent of which is set equal to the extent of the crude depth points. This resampling means that the calculation of depth as a function of lateral position is very much quicker. In areas where the depth surface is multi-valued, the deepest point is taken as this is generally thought to be the most stable.

The resampling algorithm is complex. Firstly, a grid is placed over the extent of the crude-depth points, and within each grid cell of this grid a list of the quadrilaterals that have some part of themselves inside the cell is generated. These lists then act as look-up tables that dramatically speed-up the resampling process. Given the position of a point, ***P***, the grid cell that contains it is easily calculated. Then each quadrilateral in the cell is split into two triangles and the point's barycentric coordinates calculated for each of the triangles. From the barycentric coordinates, $(\alpha, \beta, \gamma)$, which re-express the point's position as a weighted sum of the triangle's vertices, it is possible to calculate whether the point lies inside the triangle and, if so, its point of intersection.

***P*** is inside the triangle if, and only if,

$$0 \leq \alpha \leq 1, 0 \leq \beta \leq 1 \text{ and } 0 \leq \alpha + \beta \leq 1 . \qquad (2\text{-}9)$$

Once the barycentric coordinates of a point are found, they may be used to interpolate any extra properties stored at the triangle vertices, such as the depth or the dip.

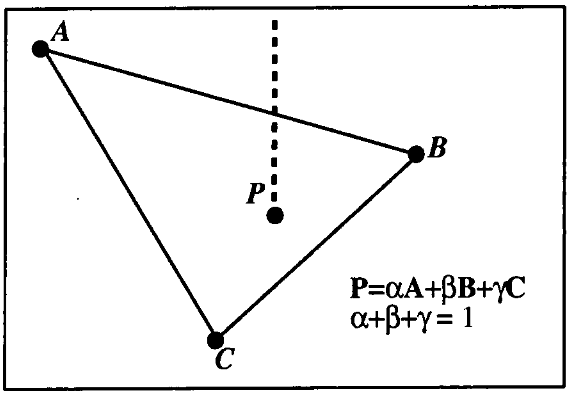


*Figure 2-21: The barycentric coordinate system*

## 2.9 Problems and Limitations with the Technique

The demigration/remigration procedure described above is limited when the time-migration velocity model is very poor or the structural complexity is too great. In these cases it is not possible to pick a consistent single-valued interpretation. Even in these complex areas, however, the techniques can still be applied. Once a velocity-depth model has been generated, the seismic data volume can be depth migrated and the depth surface found by repicking on the depth-migrated volume. A more optimal stack section for the repicked layer can then be found through modelling using normal rays, in a stage of depth demigration, and used to calculate the effect of changing the velocity model on the depth surface.

Alternatively, if the structural complexity means that stacking velocity inversion is inappropriate, more conventional focusing analysis velocity updates may be applied and the reinterpreted depth layers repicked ready to invert for the lower layers. This is a very practical and efficient procedure as it means that even in complex areas the use of more exact and considerably more expensive wave-based methods can be limited, and the initial iterations of depth migration should be much closer to the correct velocity model.

Generally, there may be some spurious crude-depth points that imply a structure that is not very geologically meaningful. These points may be due to a poorly determined migration velocity field or improperly interpreted time-migrated sections. In some cases, the migrated surfaces will need reinterpreting in some areas. This stresses the need for interaction between the geophysicist and the interpreter if good results are to be obtained.

## 2.10 Real Data Example to Map a Well from Depth into Migrated Time

It is still common practice to interpret seismic data in the time-migrated domain, and therefore it is useful to map well trajectories from the depth domain into the time-migrated domain as an aid to interpretation. This sort of analysis has great practical importance as, when tying the well to the migrated time-interpretations, the tie should not be made at the well's position in depth, but at the position in the time-migrated block. This argument is also important for techniques that require extraction of the seismic wavelet at the well position. It is clear that such a mapping requires a velocity-depth model that ties the well, i.e. the depth model should be consistent with the well information in terms of depth, dip and velocity.

To accurately map the well trajectory from the depth domain into the time-migrated domain, a two-step process is proposed (Sexton and Robein, 1996). Firstly normal rays are used to map the well trajectory from the depth domain into the stack domain, and then the time-migration process is simulated using either the Kirchhoff or finite-difference simulation depending on which algorithm was actually used. This mapping requires knowledge of the dip of the layer at the intersection of the well, and therefore mapping of the complete trajectory requires knowledge of the dip between the well markers. Dip information of this type is, in principle, available from the dipmeter log but is very high frequency and it is not a simple procedure to upscale.

Figure 2-22 shows a depth model with a well-trajectory. A series of normal rays (shown in green) is shot from the intersection of the well with the depth surfaces. The image rays from points down the well trajectory are also shown. Note that the well trajectory intersects the fourth surface in an area of fairly steep dip and therefore the normal rays cross a zone of the subsurface which is quite different from that of the image rays. It is clear, from this fact alone, that the image-ray technique will produce flawed results in this fairly typical salt dome environment. In fact, in this example, the image ray correction is in completely the wrong direction.

Figure 2-22 also shows a comparison of the four mapping techniques for the same model. In this example, the dip is only known at the places where the well crosses the depth surfaces and these points map to the coloured squares displayed in the time-migrated domain, shown in cyan for the Kirchhoff simulation and in yellow for the finite difference. The fact that these points coincide with the time-migrated interpretations from which the depth model is constructed is a good check of the consistency of the technique.

The dip dependence of the time-migrated position means that if the geological dip changes discontinuously down the well, then its trajectory in the time-migrated domain will also be discontinuous. It is also interesting to note that the Kirchhoff and finite difference techniques produce fairly consistent results in this case, as the lateral variation of the model in the area crossed by the normal rays is not too severe. Both methods produce the same kink in the well trajectory at the fourth layer, and come back close to the vertically transformed well on the fifth layer where the dip is much smaller.

Again, for comparison, the vertically transformed well is shown in red and the result of the image ray mapping is shown in green. These two simpler techniques can be used to map the full trajectory as the local dip is not needed. It is clear that for the deeper layers with appreciable lateral variability the methods can give significantly different results (over 1km for the fourth layer).

The dip of the depth horizons is obviously very important for this procedure. To check the sensitivity of the mapping to this dip, a scan can be made of a selection of dips around the one predicted by the model. The resultant stack surface can be time-migrated to produce a dip-plume, as described in section 2.5.5. If the modelling has been successful, then this dip-plume will have an area of tangency with the time-migrated interpretation for the layer. The dip-plume can also be checked against the migrated-time block to ensure the reliability of the mapping (figure 2-23).

The consistency of the mapping can be checked through use of the connectivity relations between the different seismic domains.

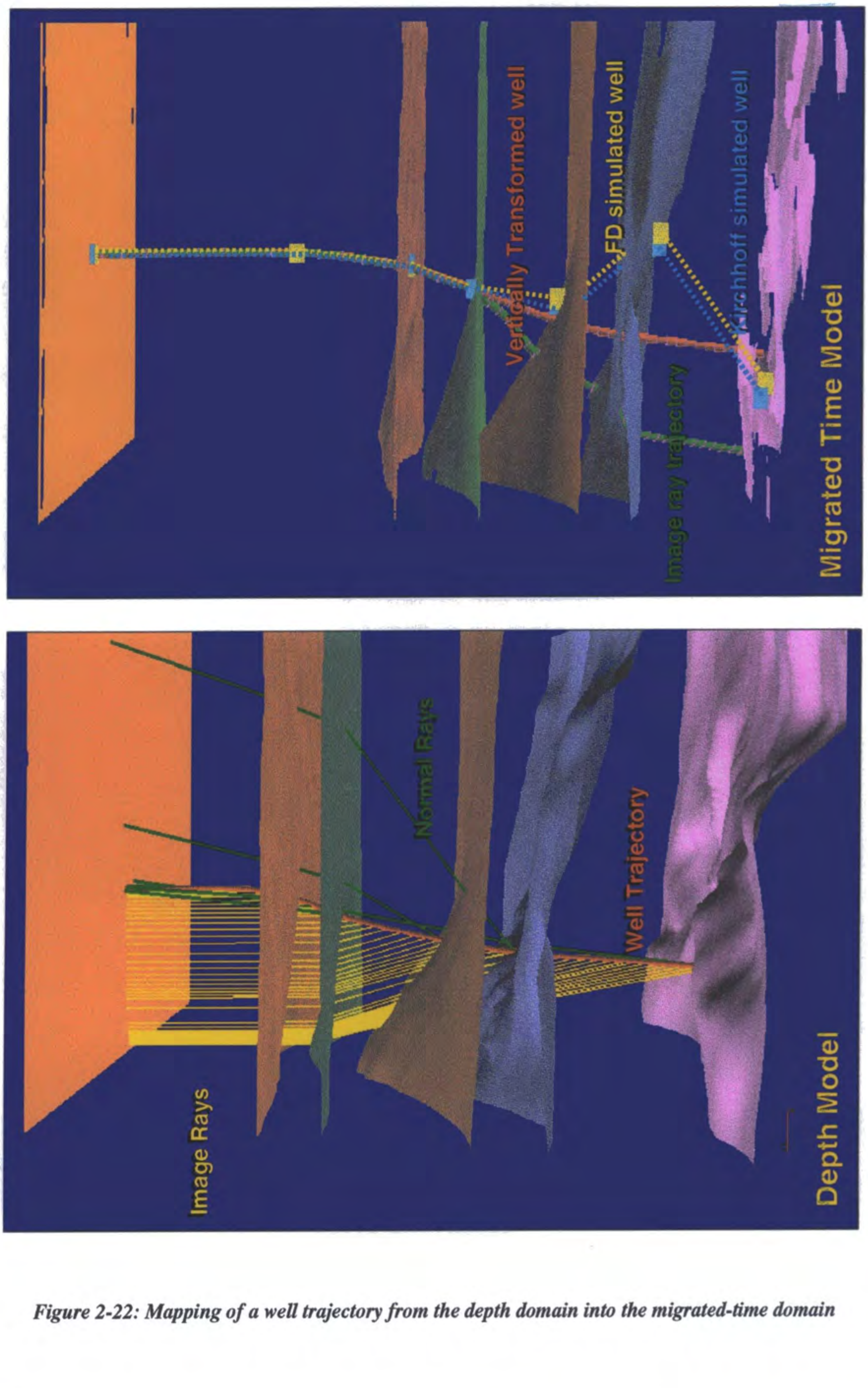


***Figure 2-22: Mapping of a well trajectory from the depth domain into the migrated-time domain***

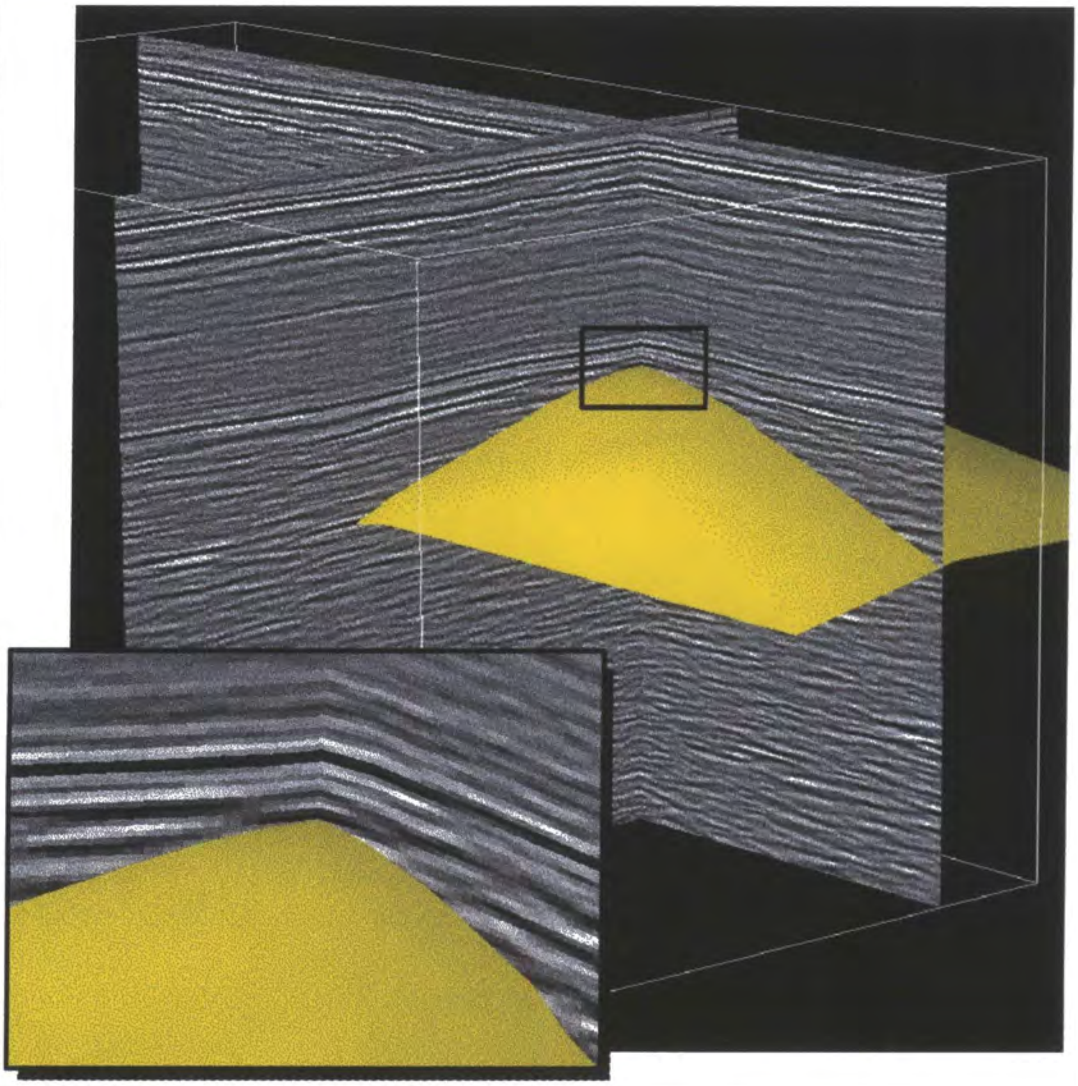


*Figure 2-23: Scanning dips and azimuths: the plume surface superimposed on to the seismic data*

# 3.0 The Direct Problem

## 3.1 Overview

Once the seismic interpretations have been converted into depth, the validity of the depth-migration velocity field must be checked. This is done by simulating as fully as possible the 3-D seismic acquisition and processing and comparing the results of this modelling with the actual data.

There are four potential criteria upon which the quality of the model can be assessed:-

- The consistency of the modelled and actual stacking velocity fields.
- The consistency of the modelled interfaces and the depth markers defined at the wells.
- The consistency of the modelled and picked well seismic traveltimes.
- The consistency of the model parameters with *a priori* knowledge based on the geology and other information from, for example, the well logs.

This chapter discusses the calculation of the first three of these factors. The use of *a priori* information is left until section 4.3.

## 3.2 Modelling the Stacking Velocity Field

To calculate a synthetic stacking velocity field, a set of offset rays must be generated to calculate the traveltime-offset curves and consequently the stacking velocity at all of the velocity analysis (VA) locations used in the inversion. The actual stacking velocity field is provided, by the processor, as a set of picks of stacking velocity against zero-offset traveltime as shown in figure 3-1. In this plot, the vertical axis is the stack time and the colour of each point represents the value of the stacking velocity. In this example, the stacking velocity picks are fairly coarse, spaced at around 500 metres with 20-30 picks at each VA location. In modern processing systems, where the stacking velocity picking is more automated, the VA locations can be far denser than this and are sometimes picked at every binning position. Other information can be associated with the picks, for example the semblance itself, or some measure of the width of the semblance peak. We then use these factors to associate confidence levels in the picks for use as weights in the inversion. The widths of the semblance peaks can be surprisingly large, sometimes of the order of several hundred metres/second, and this means that the stacking velocity field can contain considerable scatter. An understanding of these errors is essential in deciding how closely the modelled and the actual stacking velocity fields should match.

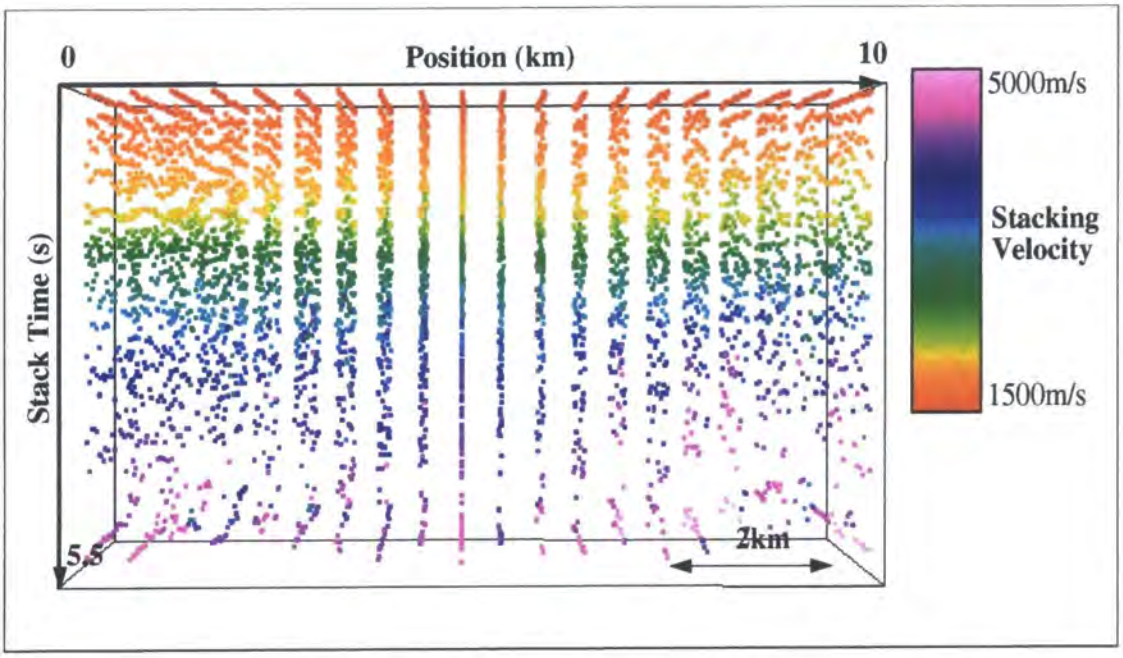


*Figure 3-1: The input to the inversion is a set of stacking velocity picks in 3D*

For the stacking velocity inversion result to be reliable, the stacking velocity picks must be made in a horizon-consistent way (i.e. the picks must be made which correspond to reflections from the interpreted interfaces used to build the model). This is because interpolation of the stacking velocity function, although common, is potentially dangerous as the variation is often significantly non-linear between picks. Fortunately, the seismic interfaces that are interpreted usually have large reflection coefficients. This means that reflections from the interpreted horizons are generally picked during stacking velocity analysis. This should never be taken for granted: the horizon consistency of the stacking velocity picks must be checked during the inversion (see section 3.4). Weights are introduced into the inversion to reduce the impact of velocity analyses which are not picked horizon-consistently.

### 3.2.1 Modelling the Zero-Offset Rays from VA Locations to Reflectors

Each model considered is the result of a depth migration and is therefore fully consistent with the time interpretations. This allows the zero-offset section produced by the demigration of the interpreted surface to be used to aid the forward modelling. After demigration, the connectivity relations of the points in zero-offset domain allows the identification of all the solutions, with their respective times and time-dips, for all the zero-offset rays for each CMP.

This is an important concept that is original to this inversion scheme and solves the non-uniqueness problem in the subsequent two-point ray-tracing. Figure 3-2 demonstrates the idea on the simple 2D synthetic from section 2.6. If we consider the VA location shown in figure 3-2, the demigrated section shows that there are three zero-offset rays from the reflector being considered to the VA location. By making use of the connectivity relationships between the points, the position of the reflecting point in depth can be accurately estimated by interpolation from the positions of the neighbouring crude-depth points, which were of course also the neighbouring points in the zero-offset section.

From these relationships, the positions of the first and last nodes of the normal ray are fixed. Any nodes on intermediate interfaces are then moved along the interfaces until Fermat's Principle is satisfied. This technique is known as ray-bending and the algorithm is described in section 3.8.1. Unfortunately, because of the discrete parameterisation of the layer geometries and the linearisation assumed by the interpolation procedure, there is no guarantee that this ray is necessary exactly a valid zero-offset ray. The reflector node is therefore unfixed and another stage of bending ensures the ray is valid.

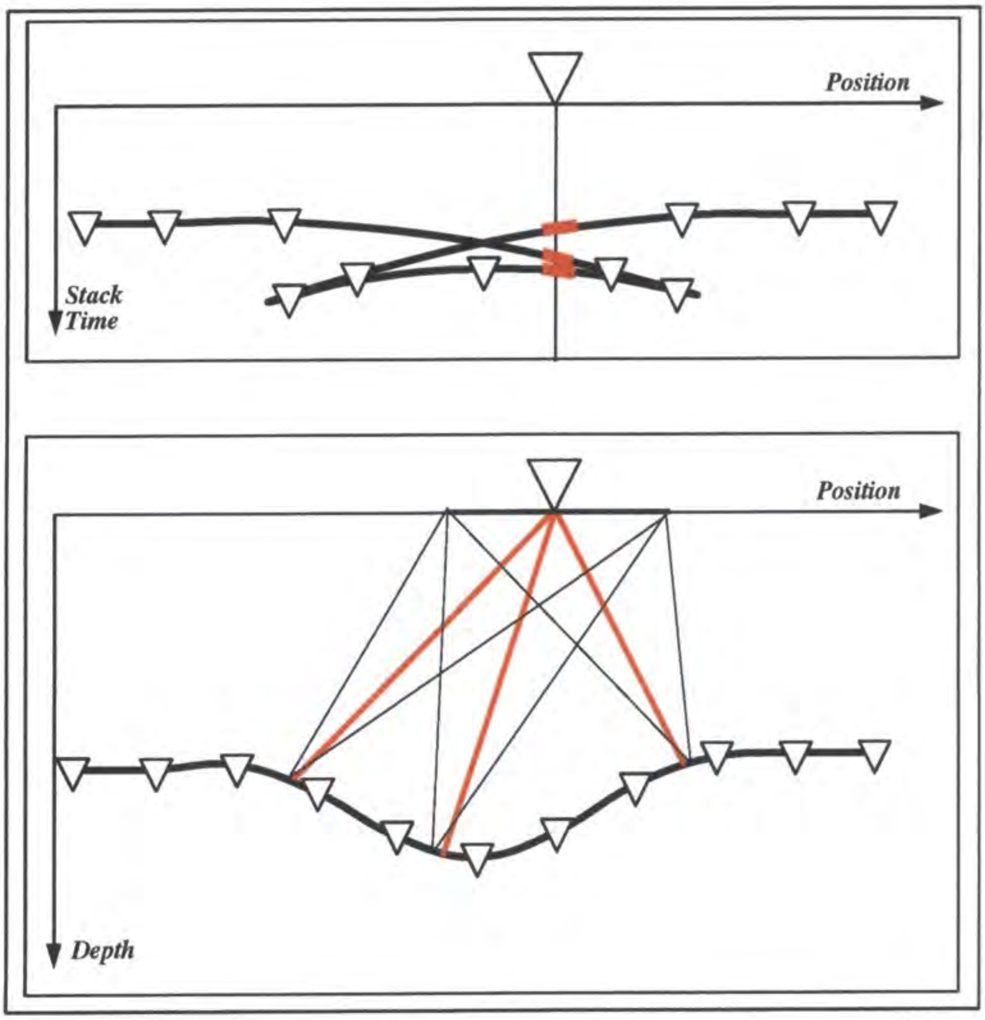


*Figure 3-2: Prediction of the zero-offset rays from the connectivity of the stack domain*

As well as the position of the reflection point, the traveltime is also approximately predicted by the stack representation. By checking that the final modelled events are consistent with the prediction, the stability of the process is assured.

This approach is also attractive because it allows the removal of reflections from fault planes if the faults themselves are not imaged, or if the stacking velocity picking off the fault plane is not considered reliable. If the fault planes are effectively removed from the migrated-time interpretation by introducing fault polygons (section 2.7.1), then they are also removed from the stack-section and the initial-guess algorithm will not predict a zero-offset ray solution. This allows precise control over the events that are modelled and drastically improves the stability of the inversion process.

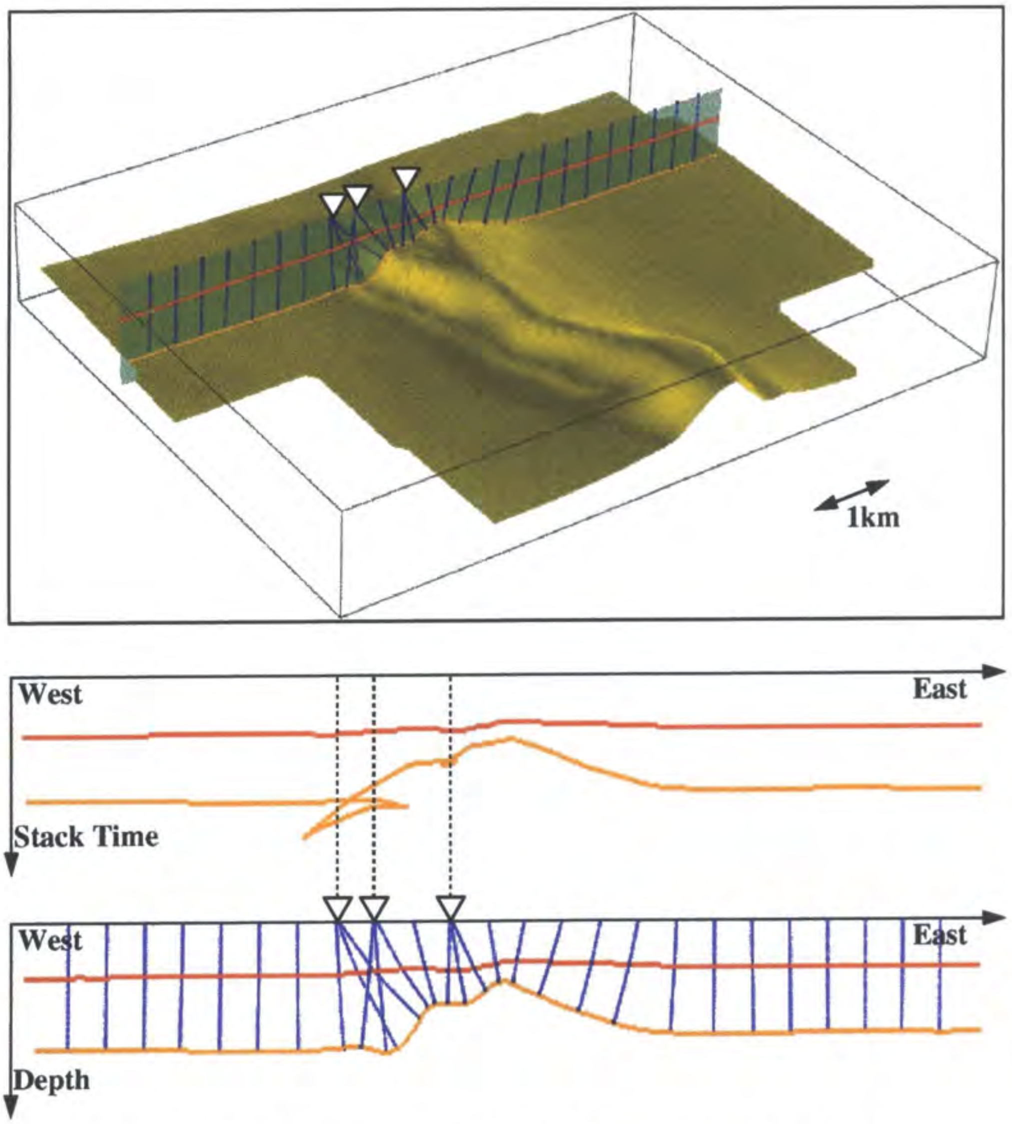


*Figure 3-3: North Sea example showing the zero-offset ray modelling*

Figure 3-3 shows this procedure applied to the real case considered in section 2.7.2. The bottom panels show west to east cross-sections for both the stack- and depth-domains. Again, these 2D displays are merely for convenience and the zero-offset rays themselves are traced in a fully 3D sense and do not lie in the plane. The VA locations with more than one solution are highlighted by triangles.

### 3.2.2 Modelling of the Offset Rays

The normal ray from a VA location to the surface can be used to define an initial estimate of the ray-path of the first of the finite-offset rays. The source and receiver positions of the offset ray are set fixed at their known positions and the reflection point is initially set equal to the reflection point of the normal ray. The intermediate nodes are then spaced evenly laterally and the depths of all the nodes fixed on their respective layer boundaries. The nodes of the ray are then moved along the layer boundaries using the bending algorithm.

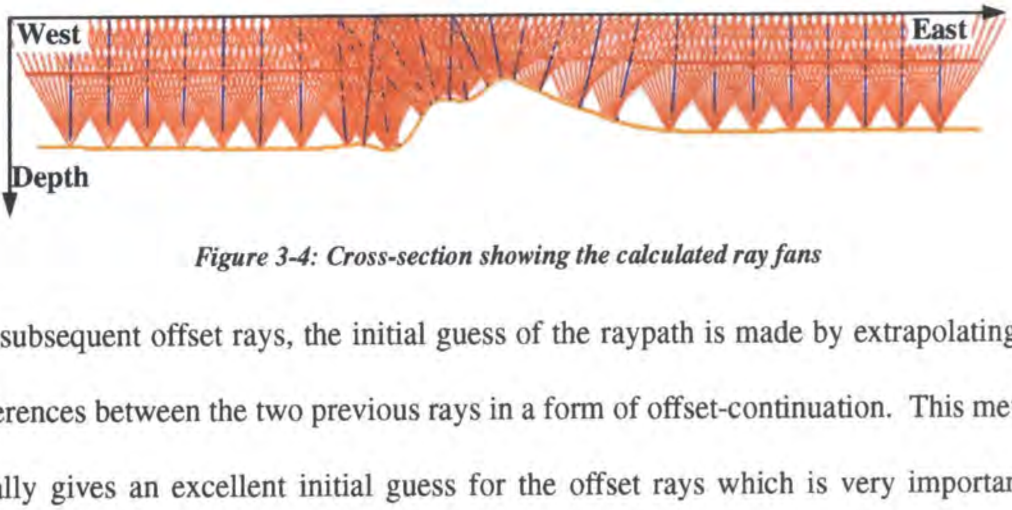


*Figure 3-4: Cross-section showing the calculated ray fans*

For subsequent offset rays, the initial guess of the raypath is made by extrapolating the differences between the two previous rays in a form of offset-continuation. This method usually gives an excellent initial guess for the offset rays which is very important to ensure both speed and stability in the ray-tracing. By using the previous rays in this fashion, the process is considerably quicker and more stable than shooting methods which implicitly calculate each ray independently of the others. Another advantage of the bending is that the source-receiver azimuth and offset can be set exactly and no subsequent interpolation is required. Figure 3-4 shows the offset ray fans calculated for the real 3D model.

## 3.3 Calculation of the Modelled Stacking Velocity from the Traveltime-Offset Curve

Once the two-point ray-tracing has generated a ray-fan, the traveltime-offset relationship can be used to calculate a modelled stacking field. When working with kinematic data, the stacking velocities are most simply calculated by least-squares regression (LSR) in the offset-squared / traveltime-squared space. This is unrealistic in that the far-offset traces have a greater influence than the short offset traces. Semblance based techniques working in the natural offset-traveltime space do not demonstrate the same bias. Also, LSR implicitly assumes that the errors are random and follow a normal distribution. This is invalid for most traveltime-offset curves where the deviation from hyperbolic moveout is clearly systematic.

Gaussian-weighted, least-square (GWLS) analysis (Thore et al., 1994) solves these problems and is also a good way of removing outliers which effect the LSR so badly. The GWLS analysis works by weighting the distance between measured and computed traveltimes by a Gaussian function with a standard deviation $\sigma$, equal to the period of the signal (figure 3-5).

The energy of summation can be expressed as a sum over offsets thus,

$$E = \sum_{i=1}^{Offsets} \exp\left(\frac{-(t_{m_i} - t_{c_i})^2}{\sigma^2}\right) \qquad (3\text{-}1)$$

and

$$t_{c_i}^2 = t_0^2 + \frac{h_i^2}{V_{stack}^2} \qquad (3\text{-}2)$$

where $t_{m_i}$,is the traveltime modelled by the ray-tracing at the $i^{\text{th}}$ offset and $t_{c_i}$, is the traveltime for a given offset, $h_i$, calculated by the normal moveout equation (equation 3-2) using the current estimates of the stacking velocity, $V_{stack}$, and zero-offset traveltime, $t_0$.

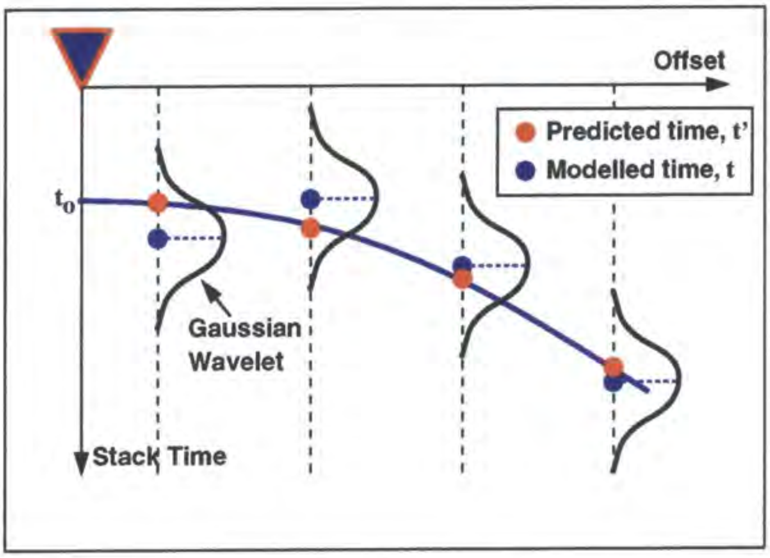


*Figure 3-5: Calculation of the stacking velocity using Gaussian weighted regression*

The optimum stacking velocity and zero-offset time are found by maximizing $E$. A Gauss-Newton inversion method is used since the partial derivatives of $E$, with respect to the stacking velocity and zero-offset traveltime, are analytic. In addition, the value of the function $E$, when normalised by the number of offsets, gives a quantitative measure of the hyperbolicity of the traveltime-offset curve, and can be thought of as a measure of stacking quality.

## 3.4 Comparison of the Modelled and the Actual Stacking Velocity Field

Once the stacking velocity at a given VA location has been modelled, it is compared with the actual stacking velocity function picked at the same location. Several parameters can be defined that quantify this comparison (figure 3-6).

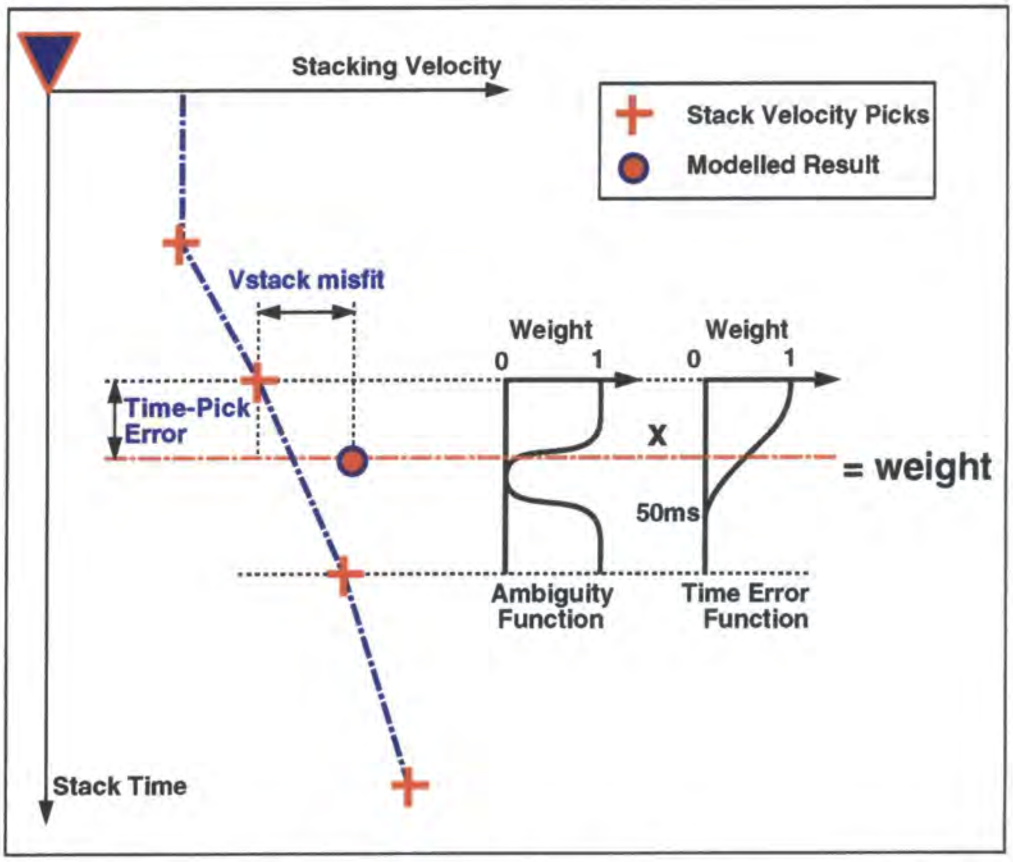


*Figure 3-6: Comparison of the modelled stacking velocity with the actual stacking velocity function*

Firstly, the time difference between the modelled time and the nearest stacking velocity pick is calculated and defines a parameter which we call the time-pick error. As the modelled zero-offset traveltime is largely independent of the depth-migration velocity (there is some dependence due to non-hyperbolicity of the traveltime-offset curve), the time-pick error is symptomatic of how horizon-consistent the stacking velocity picks are. This provides a good quality control tool to ensure the stacking velocity picks are in agreement with the time-migrated interpretations.

Once the nearest pick is found, the difference between the modelled stacking velocity and the actual stacking velocity, the $V_{stack}$ misfit, can be determined. It is this misfit that we are aiming to minimise during the inversion process.

As previously mentioned, there can be considerable noise on the stacking velocity picks as well as bad picks. This makes it very important to introduce quantitative measures of confidence in the picking at the VA location as weighting factors in the inversion. These weights are the product of two different functions. The first of these is the ambiguity function (figure 3-6), which ensures that if the modelled result is between two different picks then the actual stacking velocity is not known with certainty and therefore has a small weight. The second is a time error function, applied to reduce the impact of stacking velocity picks that were not made in a horizon-consistent way. The weight is decreased until the time-pick error exceeds 50ms. After this point, the weight is set to zero and the pick is not used in the inversion.

These properties are stored as inversion results and can be displayed painted on to the surfaces in their migrated positions (see case studies in Chapter 5 for examples).

## 3.5 Simulation of Processing Operators during the Modelling of the Stacking Velocities

During conventional processing, the final stacking velocity analysis is not performed on the raw data. A number of processing operators are applied that aim to reduce noise and improve the consistency and focusing. As these operators can drastically change the stacking velocity field, their action must be accurately accounted for during the modelling. This section describes how the effects of the most common operators can be simulated.

The simplest way of allowing for processing is to assume each process worked exactly as intended and calculate the stacking velocities for this ideal case. However, it is better to simulate the kinematics of any processing on the modelled events rather than to assume that the processing had worked exactly as intended. This is because most algorithms are approximate in that they are designed to work in idealised conditions which may not be satisfied by the model being studied. Also, to avoid introducing bias, it is better to simulate the action of the processing algorithms with the processing parameters that were actually used. This means that a number of processing parameters must also be provided as input to the inversion.

### 3.5.1 Offset Muting

Generally, shallow long offset data are excluded from the stacking velocity analysis and the subsequent stack. Long offset mutes are applied for a variety of reasons (Hatton et al., 1986) including;

- NMO stretch - an undesirable side-effect of the NMO process is wavelet distortion. The effect is at its most pronounced when the rate of change of NMO with traveltime (rather than the NMO itself) is large.
- Directivity effects.
- Dominance of non-reflected arrivals.
- NMO assumptions.
- Complex structure.

A near-offset mute must also be applied if the near-offset traces are contaminated by the direct arrival and/or refracted arrivals. It is usually a function of the water-bottom depth.

When the traveltime-offset curve is non-hyperbolic, the offset range that is used to calculate the stacking velocity is of some importance. Therefore, any muting that was applied must be simulated before the stacking velocity is calculated. A time-variant mute function can be specified with either a near- or far-offset mute. The importance of using the correct mute is very model-dependent. Figure 3-7 shows a plot of the modelled stacking velocity as a function of the maximum offset range for a flat layer at a depth of 2 km, with vertical velocity of 2000 m/s and Thomsen's (1986) anisotropy parameters, $\varepsilon = 0.2$, and $\delta = 0.05$ (details of the anisotropy parameterisation are in section 4.4.1).

These values are typical of a horizontally bedded shale where the horizontal P-wave velocity is observed to be significantly higher (20% in this case) than the vertical.

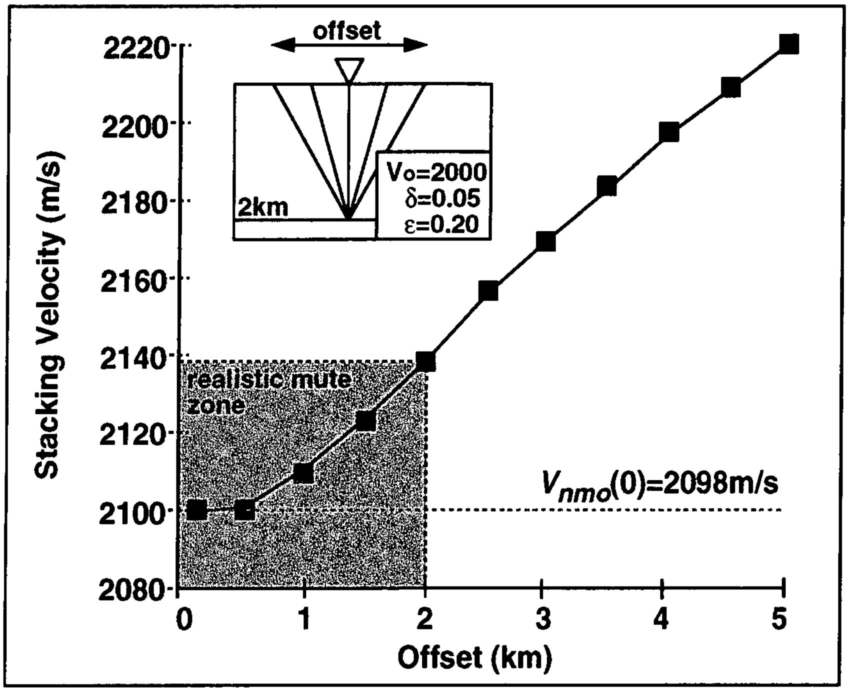


*Figure 3-7: Stacking velocity as a function of offset for a flat reflector with anisotropic overburden*

For small offsets the stacking velocity is close to $v_{nmo}(0)$ (Thomsen, 1986):

$$v_{nmo}(0) = v_0\sqrt{1+2\delta}, \tag{3-3}$$

and the stacking velocity tends asymptotically towards the horizontal velocity (2400 m/s in this case) as the offset tends towards infinity:

$$v_{stack}\Big|_{h\to\infty} = v_0\sqrt{1+2\varepsilon}. \tag{3-4}$$

This leads to the interesting idea that more information could perhaps be extracted from the stacking velocities by picking them with different mute functions. However, within practical offset/depth ratios the variation is probably not often large enough to make the differences consistently pickable. From figure 3-7 we can see that for this case the stacking velocity only varies by 40 m/s within a realistic offset/depth ratio of about 1. As the offsets continue to get longer (up to 12km in some cases!) such approaches will prove fruitful. This sort of systematic non-hyperbolic effect is a strong argument for three-term approaches, which I am sure will become increasingly important in future processing.

The source-receiver azimuth as a function of offset within the bin is also important and must be supplied as input. For marine cases, this effect is usually approximated by assuming the source-receiver pairs lie in the in-line direction. For land data the situation is more complicated as generally the source-receiver azimuth within a bin varies rapidly with offset. This variation is shown in what is called a 'spider-diagram' and must be specified for each VA location.

### 3.5.2 Dip Move Out (DMO) Simulation

Velocity analysis is now routinely carried out after dip moveout (DMO). This process attempts to remove the dip dependence of the stacking velocity field and reduce the reflection-point smear to improve the validity of the stack (Deregowski, 1986). This makes the stacking velocity field more consistent and therefore easier to pick. The DMO operator is a "migration" process that attempts to map a finite offset section into an equivalent zero-offset section and, when used in conjunction with NMO, it is referred to as a partial pre-stack migration or a migration to zero-offset (MZO).

The DMO operator rebins a data element to a different time and location and consequently DMO can considerably change the stacking velocity field. To invert a post-DMO stacking velocity field, therefore, the DMO process must be taken into account during the modelling. Consider a modelled event from source $S$ to receiver $R$ with offset $2h$, midpoint $x_0$, and traveltime $t_h$ (figure 3-8). As in a conventional seismic processing sequence, the DMO operator is sandwiched between NMO and inverse NMO corrections (figure 2-4). The NMO operators modify the traveltime and time-dip but not the binning position of the event. This means that, for accurate simulation, the NMO velocity field must also be supplied as an input. Applying NMO to the offset event, using the normal moveout velocity $V_{NMO}$, yields a 'zero-offset' time $t_0$:

$$t_0^2 = t_h^2 - \frac{4h^2}{V_{NMO}^2}, \tag{3-5}$$

from which the post NMO time-dip is readily calculated

$$\frac{\partial t_0}{\partial x} = \frac{t_h \partial t_h}{t_0 \partial x}. \tag{3-6}$$

If DMO is applied to the resulting impulse at $(x_0, t_0)$ on an NMO corrected common-offset section, with offset $2h$, the impulse response $(x_D, t_D)$ is defined (Hale, 1991) by the ellipse

$$t_D = t_0 \sqrt{1 - \left(\frac{x_D - x_0}{h}\right)^2}. \tag{3-7}$$

This means that DMO applied to a single non-zero sample of data yields non-zero samples that lie along an ellipse, as shown. This ellipse represents the DMO response for all dips from -90 to +90 degrees. Again, we can calculate the kinematic response for a given time-dip. Defining

$$x_D = x_0 + \Delta x \tag{3-8}$$

equation 3-7 becomes

$$t_D^2 = t_0^2 \left(1 - \frac{\Delta x^2}{h^2}\right). \tag{3-9}$$

Consider the DMO responses, defined by equation 3-9, of two-points separated by an infinitesimal distance from each other along the time dip-direction, i.e. $(x_0, t_0)$ and

$\left(x_0 + \delta x, t_0 + \frac{\partial t_0}{\partial x}\delta x\right)$, where $\frac{\partial t_0}{\partial x}$ is the post-NMO time-dip. The DMO corrected position of this dipping element will be the tangency point of the envelope of the two elliptical DMO operators, i.e. the position where the operators constructively interfere.

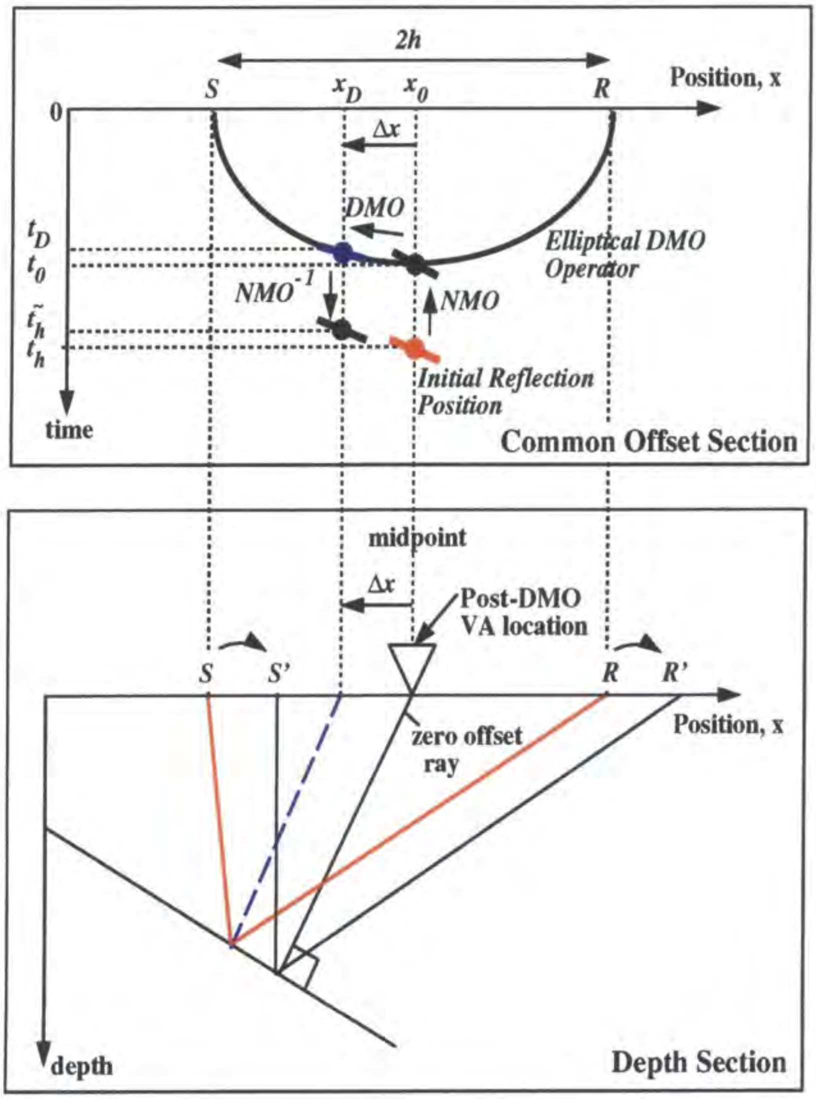


*Figure 3-8: Simulation of the DMO operator: The top panel shows how NMO, DMO and $NMO^{-1}$ operate in the common-offset domain. The bottom panel shows the corresponding rays in depth.*

The DMO displacement at this tangency point is shown to be

$$\Delta x = \frac{-t_0 \pm \sqrt{t_0^2 + 4h^2\left(\frac{\partial t_0}{\partial x}\right)^2}}{2\frac{\partial t_0}{\partial x}}, \tag{3-10}$$

where the sign of the root is chosen so as to map the event updip. This equation tells us how an event of a given time-dip is mapped by the DMO operator and can be used to simulate the effects of the DMO operator on the modelled stacking velocity field (Robein et al., 1995).

Finally, the traveltime is corrected with an inverse NMO to give

$$(\tilde{t}_h)^2 = t_D^2 + \frac{4h^2}{V_{NMO}^2} . \tag{3-11}$$

It is this traveltime, $\tilde{t}_h$ , that is used to calculate the post-DMO modelled stacking velocity.

These equations allow us to simulate precisely the DMO procedure (bottom panel of figure 3-8). Firstly, the zero-offset ray is traced from the VA location to the reflector. This zero-offset event is unaffected by DMO. For the first offset ray, the source and receiver locations are placed symmetrically around the zero-offset ray, as shown, and the internal nodes moved by the bending algorithm. The reflection point of this offset ray is displaced updip from the reflection point of the normal ray fired from the midpoint of the source-receiver pair. This offset ray, which represents an event with a given offset, time and time-dip, then has the DMO operator (equation 3-10) applied to it. The DMO operator attempts to rebin the offset ray to a normal ray, shown here as a dashed line, with the same reflection point. It is clear that this normal ray hits the surface some distance away from the midpoint. The DMO operator gives us the binning displacement relative to the event's source-receiver midpoint: in this case, a distance $\Delta x$ to the left.

When we are calculating a post-DMO stacking velocity field we consider all the events that are binned at the VA location after the DMO operator has been applied. It is clear that this event, with source and receiver equidistant from the zero-offset ray will not be binned at the VA location and therefore does not contribute to the stacking velocity at the VA location. What we require, therefore, is not a CMP gather as in conventional stacking velocity inversion, but instead what we term a pre-DMO gather, i.e. the set of events that will be binned at the VA location after the DMO process has been applied.

The construction of a pre-DMO gather is iterative. The source and receiver positions are moved a distance $\Delta x$ to the right (to $S'$ and $R'$) and the internal ray nodes re-bent. The DMO operator is then iteratively reapplied to this event and any residual corrections used to further update the source and receiver positions. In each step, the source and receiver positions are moved by the same amount and in the source-receiver plane. This means that the source-receiver offset and azimuth remain constant. The process converges rapidly.

In a simple case where the overburden velocity is constant, the pre-DMO gather will also be a common reflection point, or common-image gather. In this ideal case, the traveltime after DMO, but before inverse NMO, will also equal the traveltime of the normal ray. Generally, as will become apparent in the examples in section 3.5.3, this will not be the case and there will be residual reflection point smear.

This approach is repeated for each offset that is being modelled, although for speed the initial guess of subsequent offset rays uses the offset-continuation principles explained in section 3.2.2. The initial guess, therefore, includes a large proportion of the DMO correction before the process starts, and this makes this approach both stable and very quick to apply. Figure 3-8 is in 2D only, but the approach is fully implemented in 3D. This extension is trivial, however, as the conventional DMO operator is a 2D operator that operates in the source-receiver plane.

### 3.5.3 Examples of DMO Simulation

To demonstrate the accuracy of the simulation procedure, a synthetic example (figure 3-11) with two dipping layers was considered. The dips are 20° and 30° and the velocities are constant within each layer at 2000 m/s and 3000 m/s. Common-midpoint gathers were generated by ray-tracing in both dip and strike directions. Figure 3-10 shows a close-up of the gather from the second reflector: the up-dip reflection point smearing, in the dip direction, is clearly visible. Stacking velocities were then calculated from the traveltime-offset curves.

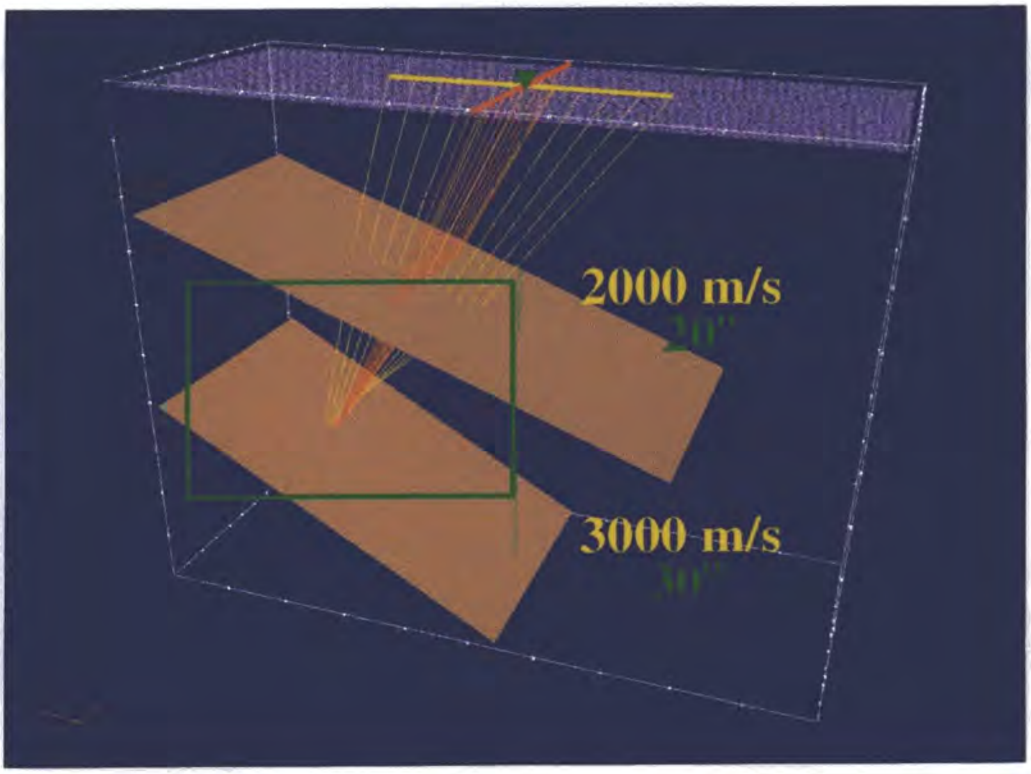


*Figure 3-9: Synthetic two-layer model showing dip and strike modelling of CMP gathers*

The stacking velocity for reflector 1 in the dip direction is calculated as 2128 m/s. This is in perfect agreement with Levin's (1971) formula which gives the stacking velocity, $V_{stack}$, for a homogeneous model of propagation velocity, $V$ and reflector dip, $\theta$.

$$V_{stack} = \frac{V}{\cos\theta} \tag{3-12}$$

*Figure 3-10: Close-up of the dip and strike CMP gathers for the second layer*

*Figure 3-11: Close-up of the dip and strike pre-DMO gathers for the second layer*

For reflector 2, the stacking velocity in the dip direction is 2734 m/s, much higher than the stacking velocity in the strike direction (2420 m/s). The stacking velocity in the strike direction is close to the root mean square (RMS) velocity along the normal ray (2440 m/s).

This dip and strike modelling was repeated after DMO simulation. As expected, for the first layer the reflection-point smear is effectively removed, and the stacking velocity calculated from the traveltime-offset regression is exactly 2000 m/s. For the second reflector, the reflection point is now smeared down-dip (figure 3-11), but the smearing is considerably smaller than in the case where no DMO is simulated. Consequently, the modelled stacking velocity is 2335 m/s, which is much lower than either the RMS velocity down the normal ray or the strike stacking velocity which are 2440 m/s and 2420 m/s, respectively. Even for this relatively simple case, DMO over-corrects the stacking velocity by around 4%, and the post-DMO stacking velocities cannot be assumed to be equal to the RMS velocity along the normal ray.

To check the accuracy of this simulation, a synthetic 2D line in the dip direction was calculated using a fully dynamic ray-tracing package different from the one used to generate the offset fan. These synthetic data were then processed in the conventional way with an NMO correction followed by a F-K constant velocity DMO, inverse NMO and finally stacking velocity analysis at the VA location being considered.

The results of this post-DMO stacking velocity analysis are shown for the first and second reflectors in figure 3-12 and figure 3-13. The right panel of these figures is the semblance plot; the left shows the NMO corrected gather for the maximum semblance pick.

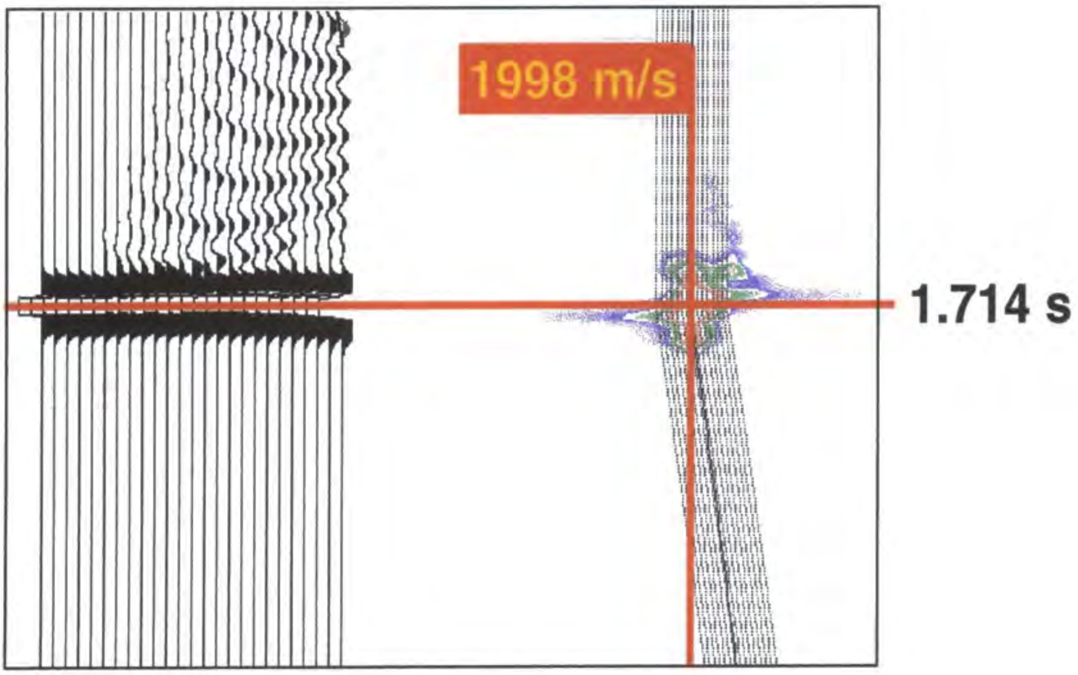


*Figure 3-12: Semblance analysis on the first reflector after DMO*

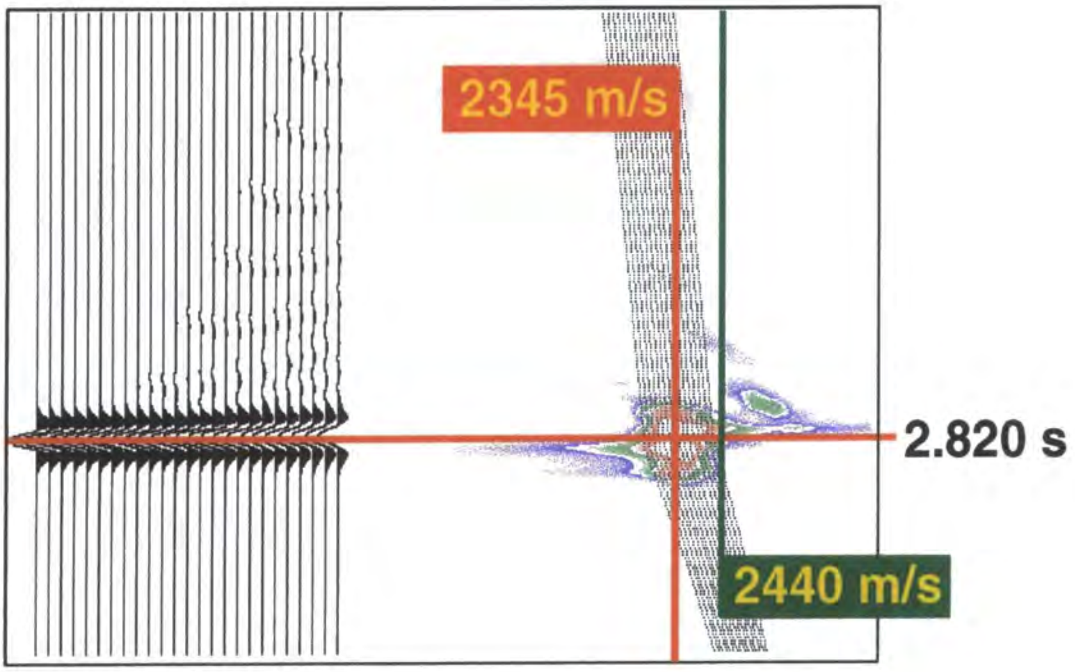


*Figure 3-13: Semblance analysis on the second reflector after DMO*

For the shallow reflector the DMO operator has worked perfectly: the effect of dip on the stacking velocity has been removed and the stacking velocity simply equals the interval velocity in the first layer. For the second reflector, however, the stacking velocity is estimated as 2345 m/s. The RMS value down the normal rays is also shown on the semblance plot in figure 3-13, and it is very clear that the semblance peak is some 100 m/s lower. The difference between the simulated stacking velocity and the result of the real processing is only around 10 m/s, which is negligible given the width of the semblance peak during the velocity analysis. This result, therefore, confirms the simulation technique can be used to accurately predict post-DMO stacking velocities.

The results are summarised in the table below:

| | **Vstack dip after DMO simulation (m/s)** | **Vstack dip after DMO processing (m/s)** | **$V_{RMS}$ down the normal ray (m/s)** |
|---|---|---|---|
| Reflector 1 | 2000 | 1998 | 2000 |
| Reflector 2 | 2335 | 2345 | 2440 |

**TABLE 1. Comparison of DMO simulation and processing**

There are two main conclusions that can be drawn from this simple example. Firstly, DMO does not correctly remove the effects of dip from the stacking velocity field and the difference from this ideal behaviour is large enough to be picked during velocity analysis. Secondly, the results of the simulation are in excellent agreement with the processed values and confirm that the simulation technique can be used to accurately predict post-DMO stacking velocities.

Figure 3-14 shows a comparison of the modelled stacking velocities with and without DMO simulation on two layers of a 3D model. The stacking velocities have been projected down the normal ray on to the depth surfaces and colour coded. This type of display provides a useful diagnostic because it shows the stacking velocity field in its migrated position.

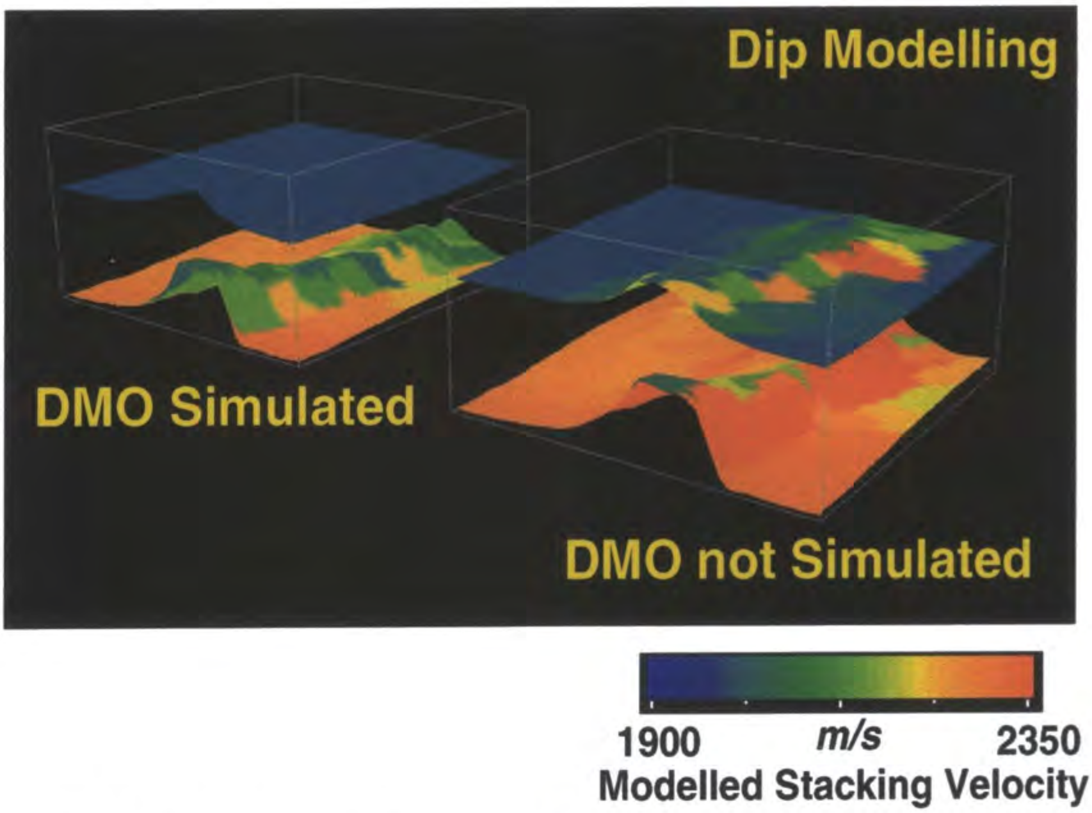


***Figure 3-14: Comparison of the modelled stacking velocity with and without DMO simulation***

Within the first layer the interval velocity was set to be constant at 2000 m/s. The panel on the right shows the stacking velocities without DMO simulation. As expected, the stacking velocities are clearly seen to increase in the areas of dip as predicted by Levin's (1971) formula. With DMO simulation, this effect is perfectly removed and the stacking velocities are equal to the interval velocity everywhere. For the second layer, the results are more unpredictable. In this case the velocity overburden is not constant due to the structure of the first layer and a vertical gradient in the second. The constant velocity DMO does not work correctly.

Figure 3-15 shows a close up of the pre-DMO ray fans on one of the flanks of the second layer. The residual reflection-point smear on the pre-DMO gathers is clearly visible. The extent and indeed the direction of this smear is dependent on the local structure and highlight the necessity of an accurate simulation.

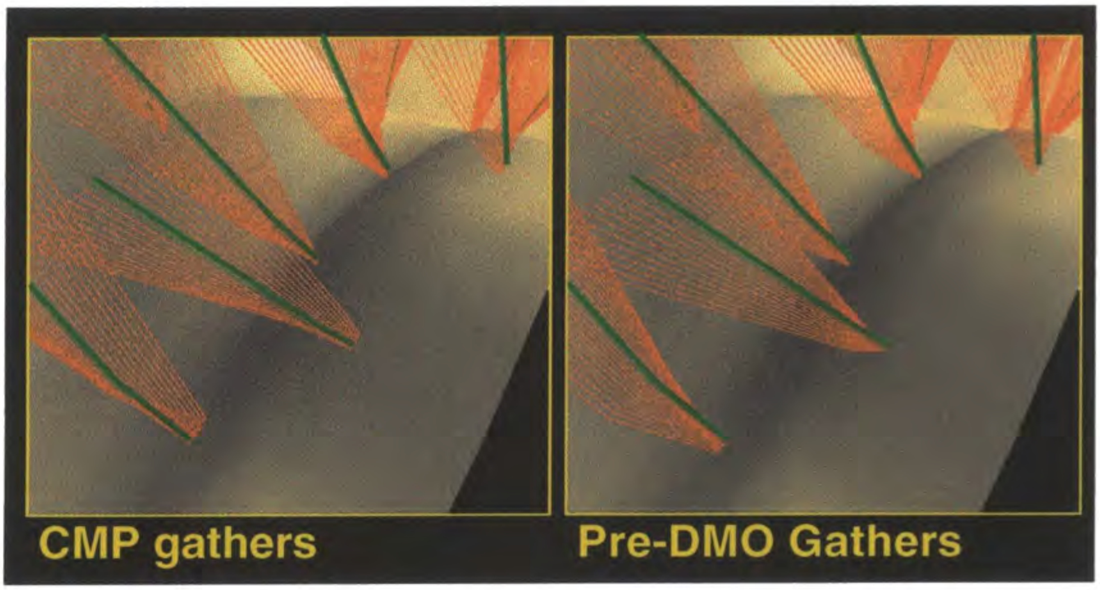


*Figure 3-15: Comparison of CMP and pre-DMO gathers on the flank*

### 3.5.4 Simulation of Pre-Stack Time Migration (PreSTM)

Another processing-step which is becoming increasingly popular is pre-stack time-migration, or 3D PreSTM. The method was first introduced by Marcoux et al. (1987) under the name of MOVES. The procedure is assumed to improve the (time) imaging of steep dips and at the same time to yield more accurate velocity information. This is because DMO removes some of the dip effects on the stacking velocities and the time-migration brings the reflections to a more correct position in space. As a consequence, stacking velocities are more accurately picked, which is of obvious benefit to the inversion. The stacking velocities should also be close to vertical RMS velocities and therefore produce better models when used in the inverse Dix formula.

The assumptions inherent in the process are strictly only valid for homogeneous isotropic medium. In a similar way to the DMO operator, we have developed a model-based technique which accurately simulates the kinematics of PreSTM to predict the traveltime as a function of offset in a heterogeneous, anisotropic 3D earth model. The advantages of PreSTM may then be evaluated in any realistic situation.

PreSTM is a four-step process. Firstly, NMO-DMO is applied. This attempts to map each common offset section into a zero-offset section. Secondly, each of the resultant 'zero-offset' sections (one per offset) is time-migrated using a 3D post-stack algorithm. Thirdly, the initial NMO is removed and a conventional stacking velocity analysis is carried out before a final NMO, stack and residual migration are applied to the whole dataset. The simulation procedure (Robein et al., 1997) faithfully mimics the kinematics of this sequence and is applied in exactly the same way as the DMO simulation. The time-migration operator is simulated by applying the Kirchhoff operator, as described in section 2.5.1.

When simulating PreSTM, the initial estimate of the normal rays is no longer accurately estimated by the procedure described in section 3.2.1. Instead, the demigrated zero-offset section must be re-time-migrated, although not with the final time-migration velocity model that was used in the demigration but with the time-migration velocity model that was used in the PreSTM (figure 3-16). In a similar way to before, this time-migrated section can be analysed to find the number and positions of events that map, not to a given zero-offset position as before, but to a given VA location after PreSTM. Moreover, the connectivity relationships between the domains give the position of this event in both the stack- and depth-domains and therefore, as before, both ends of the zero-offset ray can be accurately estimated and the solution to the problem is stable.

If the PreSTM positions the events perfectly, although in general this will not be the case, then the reflection point will be vertically below the VA location. Another important point is that because the velocity analysis is carried out in the time-migrated domain, there is much less likelihood of multiple solutions, which can interfere and make picking the stacking velocity function problematic.

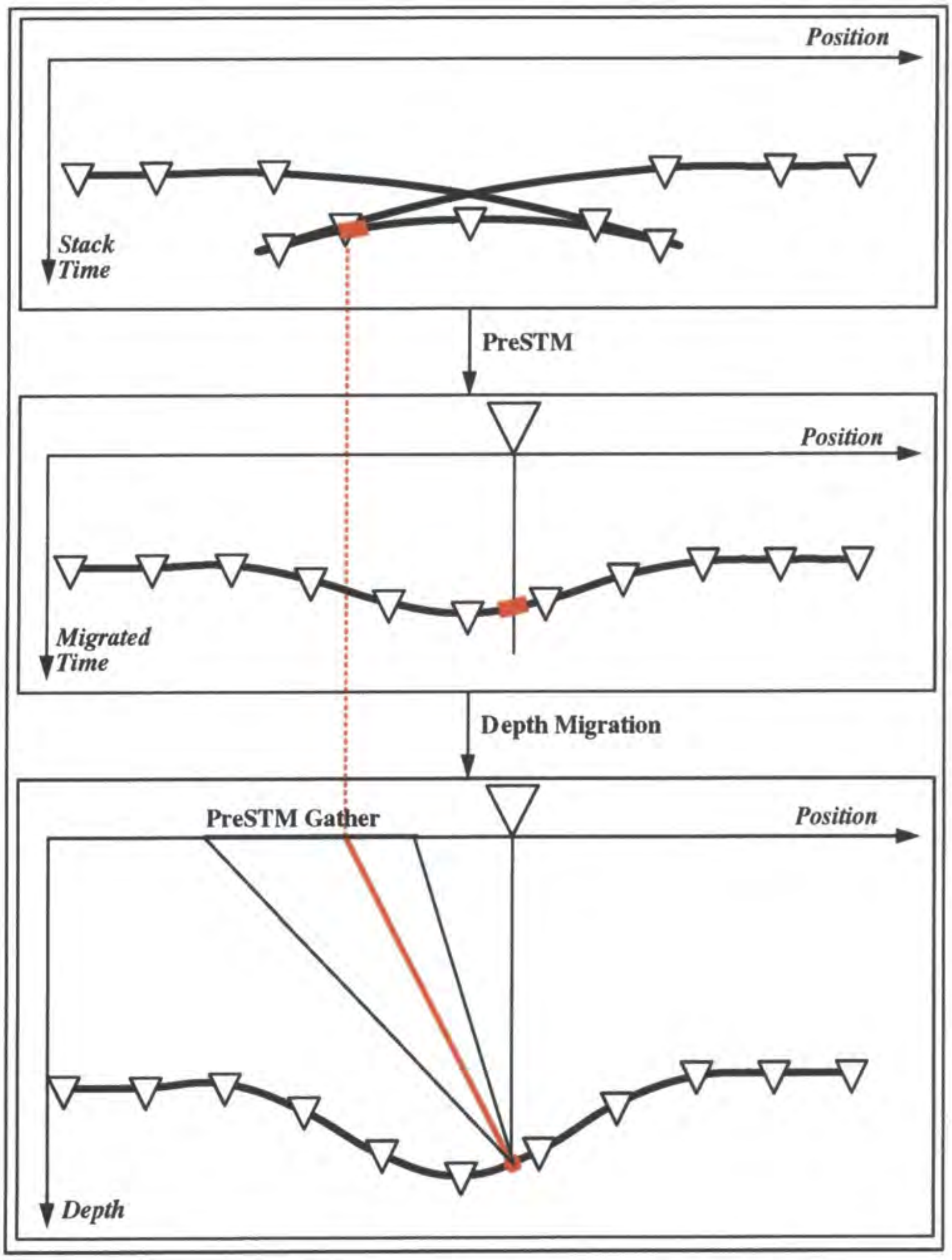


*Figure 3-16: Estimation of the zero-offset rays when velocity analysis follows PreSTM*

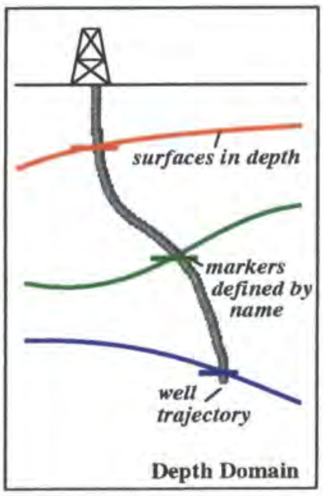
## 3.6 Comparison with Well Marker Depths

Calibration of the seismic data with the wells is a complicated and messy problem. Generally, edited sonic and density logs can be used to calculate acoustic impedance logs. A time-depth curve is calculated from the integrated sonic log, and this is adjusted to tie with the check-shot survey or vertical seismic profiling (VSP) data. Given this time-depth relationship, the acoustic impedance log is converted into vertical time.

Comparison of the 3D seismic data with the logs requires knowledge of the seismic wavelet. This is either assumed to be zero-phase with some specified frequency content, or more commonly nowadays is found by inversion (seismic to well tie). Given an estimate of a wavelet, a synthetic seismogram can be estimated at the well position by convolution with the acoustic impedance log assuming a 1D convolutional earth model. The error in the wavelet is calculated by correlation of the synthetic seismogram with the seismic data in the vicinity of the well. The phase and amplitude characteristics of the wavelet can then be iteratively modified in order to minimize this error.

After checking the character, and, if more than one well is present, the lateral continuity of the resultant wavelet, the best-fit synthetic seismogram is used to identify the main seismic events. Correlation of the 3D seismic data with the synthetic seismogram then allows calibration and identification of the seismic horizons. The interpreted horizons can thus be associated with a vertical time on the synthetic seismogram. The time-depth curve at the well is then used to map this point back into depth. In this way, the depth of the interpreted horizons is estimated at the wells.

The well markers, therefore, have two main sources of error. The first is in the knowledge of the position of the well trajectory in depth. This is thought to be very small in modern acquisition systems. The second, considerably larger, error is the association of the seismic horizons with the logs. There are many potential sources of error due the assumptions made in method described above, and the quantification of these errors is very case-dependent.



The input into the inversion is a set of well markers that are associated with given interpretations using the horizon name. The full well trajectory can also be defined for visualisation and mapping the well into migrated-time using vertical stretch or image ray techniques, which do not require the local time dip.

*Figure 3-17: Definition of wells within the inversion*

## 3.7 Calculation of Well Seismic Rays

If well seismic traveltimes have been picked, then this information can also be modelled to help constrain the inversion. The ray-tracing is flexible enough to model reflections and even multiples as well as direct arrivals, although it is not as common for these events to be picked from the well seismic data. The ray-tracing can also be used in an interactive way to help identify some of the more complex events.

Often three-component geophones are used, and so mode-converted arrivals can also be considered as these are more commonly observed from the well seismic data, especially P-S reflections. Any type of well seismic acquisition can be included although offset VSP's and walkaway surveys are most common. An extra interesting feature of this sort of modelling tool is that it allows the resolution of different model parameters to be tested as a function of acquisition design.

Practically, the inversion is provided with a set of source positions and receiver positions and a look-up table, specifying which receivers were live when a given source was fired. Each source-receiver pair also has a picked traveltime associated with it. The interval in which a given sub-surface receiver lies is a function of the current model.

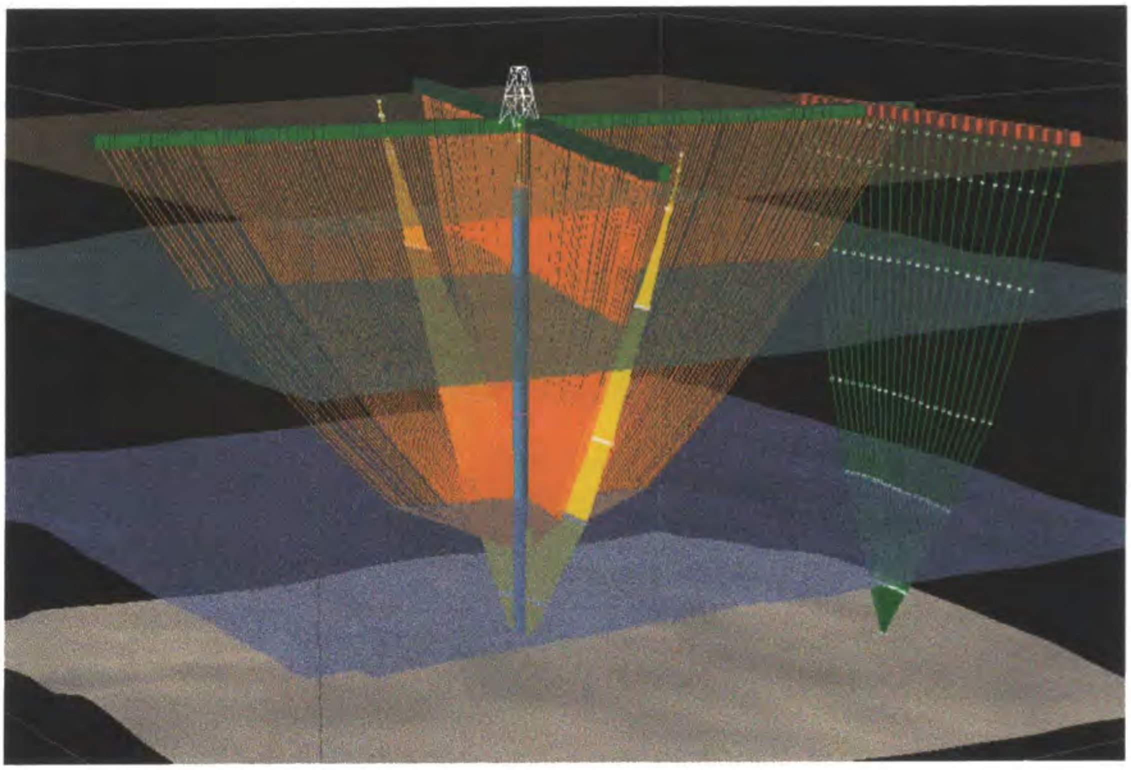


*Figure 3-18: Calculation of well-seismic rays*

Figure 3-18 shows an real data example of well seismic modelling. In this case there are two walkways at right angles to each other, shown in orange, and two offset vertical seismic profiles (VSP) and a vertical check shot, shown in yellow. A stacking velocity offset fan is also shown.

## 3.8 Ray-Tracing

Due to the specialised nature of the problems being solved, in particular the offset ray-tracing to model the post-processed stacking velocity field, all of the algorithms used in the inversion were developed from scratch. There are two basic ray-tracing algorithms that are used. The first is a shooting algorithm. Given a starting point and an initial angle of propagation, a ray is traced until a given interface is reached or a traveltime is exhausted. This function is used for the depth migration code and to calculate the Frechet matrix derivatives during the inversion loop. The second is a bending algorithm that, given an initial guess at a ray-path, will adjust the ray until Fermat's principle is satisfied. This algorithm can be used to solve the two-point problem, as needed by the stacking velocity modelling and the well seismic ray modelling, or to find zero-offset rays from a given reflector when the initial propagation direction is not known.

Given the macro-layer definition of the velocity model, the ray-path can be uniquely defined by its intersection with the layers (figure 3-19). These intersections are referred to as nodes and are shown as yellow circles. The problem of ray-tracing is the movement of these nodes, constrained to lie on the depth surfaces, in order to satisfy Snell's law. As the interfaces are not described analytically and the anisotropic solution of Snell's law is transcendental anyway, the problem must be solved by an iterative approach.

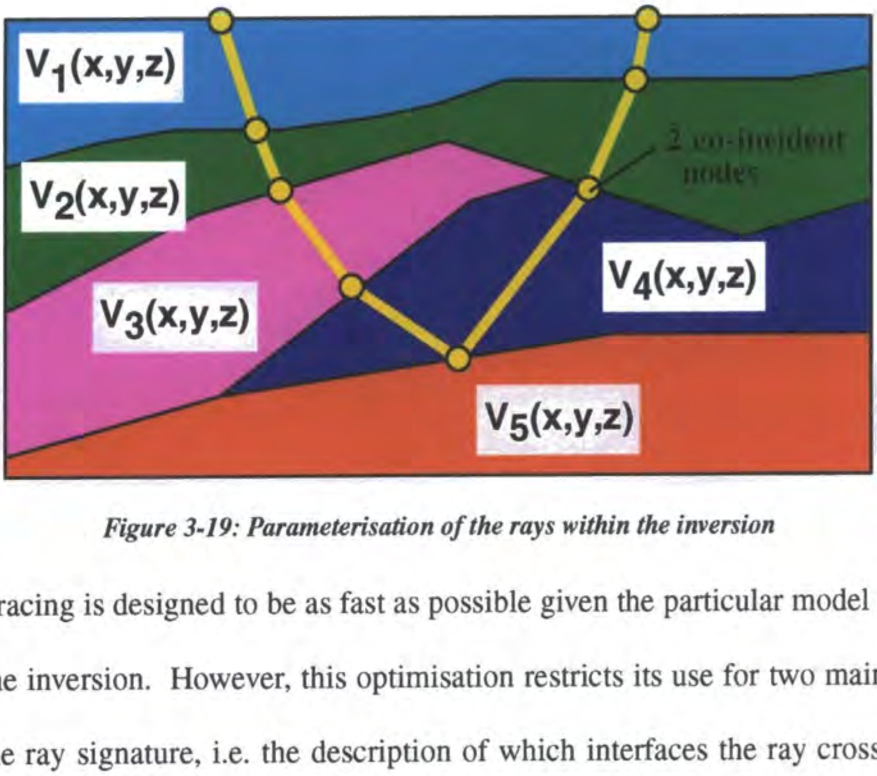


*Figure 3-19: Parameterisation of the rays within the inversion*

The ray-tracing is designed to be as fast as possible given the particular model definition used in the inversion. However, this optimisation restricts its use for two main reasons. Firstly, the ray signature, i.e. the description of which interfaces the ray crosses and in which order, must be pre-defined in both the shooting and the bending algorithms. For the ray shown in figure 3-19 there are 9 nodes, although two are coincident, and the signature is given by $[-1, 0, 1, 2, 3, 2, 1, 0, -1]$. This limits the use of the algorithm in cases where the layer geometries are very complex or multi-valued because the ray signature is not known beforehand as it can change in subsequent iterations. Pinch-outs, however, as in figure 3-19, and velocity lenses can be handled by layers of zero thickness.

Secondly, the ray-tracing is extremely rapid because the velocity variation between two nodes is assumed to be linear with constant anisotropy parameters. This approximation is not as restrictive as it seems at first because the lateral velocity variations can still be catered for with a high degree of accuracy, so long as the wavelength of these variation is such that the velocity variation over any ray segment is approximately linear. When non-linear lateral velocity variations exists, the ray-tracing works by first calculating a local gradient over the extent of the ray segment. Once the ray has been calculated, the traveltime calculation can perform a numerical integration to more accurately calculate the traveltime allowing for higher order velocity variations if this is deemed necessary.

### 3.8.1 Bending: The Two-Point Problem

The traveltime, $t$, as a function of the ray node positions, $\boldsymbol{r} = (x, y, z)$, can be approximated by its Taylor series expansion around a given position, $\boldsymbol{P}$:

$$t(\boldsymbol{r}) = t(\boldsymbol{P}) + \sum_i \frac{\partial t}{\partial r_i} r_i + \frac{1}{2} \sum_{i,j} \frac{\partial^2 t}{\partial r_i \partial r_j} r_i r_j + \dots \qquad (3\text{-}13)$$

Assuming the function $t$ can be locally approximated by a quadratic form, we have

$$t(\boldsymbol{r}) \approx c - \boldsymbol{b} \cdot \boldsymbol{r} + \frac{1}{2} \boldsymbol{r} \cdot A \cdot \boldsymbol{r}, \qquad (3\text{-}14)$$

where

$$c \equiv t(\boldsymbol{P}), \qquad (3\text{-}15)$$

$$\boldsymbol{b} \equiv -\nabla t|_P, \qquad (3\text{-}16)$$

and

$$[A]_{i,j} \equiv \left. \frac{\partial^2 t}{\partial r_i r_j} \right|_P . \tag{3-17}$$

The matrix $[A]_{i,j}$, whose components are the second partial derivatives of the function, is called the Hessian matrix of the function at point ***P***.

In the approximation, the gradient of *t* is easily calculated as

$$\nabla t = A\boldsymbol{r} - \boldsymbol{b} . \tag{3-18}$$

This implies that the gradient will vanish - the function will be at an extremum at a value of $\boldsymbol{r}$, obtained by solving the equation

$$A\boldsymbol{r} = \boldsymbol{b} . \tag{3-19}$$

Using an iterative approach, this allows the stationary points of the ray-path's total traveltime to be found and thus the ray-paths that satisfy Fermat's principle. Once expressions have been found for the traveltime and its first and second derivatives in terms of the node positions, the correct ray can be found rapidly using a damped Gauss-Newton inversion scheme. As the movement of a given node only effects the traveltimes through adjacent ray segments, the Hessian matrix is symmetrical and block triagonal. Consideration of these symmetry properties means that the amount of calculation required to calculate the inverse is drastically reduced, and the solution of the bending problem is found simply by row elimination.

In complex structure, the approximation of the traveltime being locally quadratic with respect to the node positions can be significantly invalid. This leads to instability during the inversion and requires that the inversion be damped to prevent problems. Practically, this damping is achieved by adding a term to the diagonal elements of the Hessian. By carefully defining the initial ray estimates, the speed and stability of the ray-tracing is dramatically improved.

### 3.8.2 Shooting

As the geometry is not defined analytically, the shooting problem is essentially solved by bending in a given layer. On entry to the algorithm, a ray signature is defined. The first node of the ray is fixed and initial traveltime slowness vector (or shooting angle) at this node supplied as input. The calculation then proceeds node by node. The second node is initially placed vertically below (or above) the first on the layer boundary dictated by the signature. Then the three-component slowness, due to this vertical segment, is calculated at the position of the fixed node (see section 3.9). This slowness value is compared with the required slowness, supplied as input, and the position of the second node iteratively moved in order to minimize the discrepancy.

Once this node position is found it is fixed, and the slowness vector in the next layer is calculated by calculating the slowness in the new velocity field. This calculation allows for the refraction at layer boundaries using Snell's law, which simply states that the tangential component of the traveltime gradient to the interface is continuous across it. Subsequent nodes are then calculated in the same way. The node movements are initially found using a steepest descent algorithm. When close to the solution, however, this is modified to include the second-order Hessian terms. Usually, this converges within three or four iterations.

As well as shooting through a pre-defined model, this algorithm is also used for the map migration (section 2.3). In addition to the initial ray slowness, a one-way traveltime is supplied as input (the demigration operator, section 2.5, defines both). The first node is fixed at the surface and the ray traced through the velocity model as before. When the last interval is reached, however, the final interface is not yet defined. The slowness in the last interval and the remaining traveltime for the zero-offset ray are then used to analytically calculate the crude depth point at the end of the ray. If there are lateral velocity variations in the last layer, then an initial lateral gradient is estimated, the ray traced, a new-lateral gradient estimated using the resultant ray-path, and the process iterated until the ray-path is consistent with the velocity variation along it.

## 3.9 Time and Time-Derivative Calculations

### 3.9.1 Isotropic Case

The raypath through an isotropic medium with a velocity gradient (not necessarily vertical) is a circle in the plane of the gradient vector, $\boldsymbol{k}$, and the displacement vector between its end-points, $\boldsymbol{d}$.

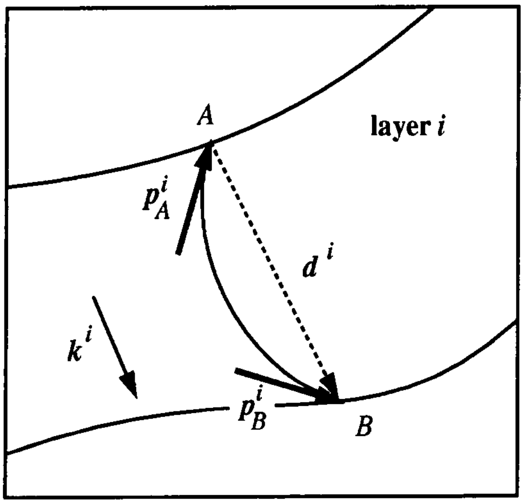


*Figure 3-20: Calculation of the ray slowness through a gradient media*

The ray-path lies in the plane of $\boldsymbol{d}$ and $\boldsymbol{k}$ and so the three-component slowness vector, $\boldsymbol{p}$, at any point along the ray, is also in this plane and can be written as

$$\boldsymbol{p} = \alpha(\boldsymbol{d} + \beta\boldsymbol{k}), \tag{3-20}$$

where $\alpha$ and $\beta$ are scalars that must be determined.

The magnitude of the slowness is the reciprocal of the instantaneous velocity,

$$|p| = \frac{1}{v}. \tag{3-21}$$

As there is no velocity variation in the direction perpendicular to $\underline{k}$ the slowness in this direction is conserved, and so

$$p_A - (p_A\hat{k}) \cdot \hat{k} = p_B - (p_B\hat{k}) \cdot \hat{k}. \tag{3-22}$$

As mentioned earlier, the ray-path is circular, and therefore

$$\hat{p_A}d = \hat{p_B}d. \tag{3-23}$$

From these constraints, expressions can be found for the slowness at A and B which are given (e.g., Williamson and Raynaud, 1995) as;

$$p_A = \frac{d + \frac{|d|^2}{2v_A}k}{v_A\left|d + \frac{|d|^2}{2v_A}k\right|} = \frac{a}{v_A|a|}, \tag{3-24}$$

and

$$p_B = \frac{d - \frac{|d|^2}{2v_B}k}{v_B\left|d - \frac{|d|^2}{2v_B}k\right|} = \frac{b}{v_B|b|}, \tag{3-25}$$

which reduce to the more intuitive form $p = \frac{d}{v|d|}$ when the gradient is small.

The total traveltime change due to the movement of given node is simply the sum of the traveltime change in the adjacent ray segments.

$$\frac{\partial T}{\partial \boldsymbol{r}_b} = \nabla_b T = \boldsymbol{p}_b^i - \boldsymbol{p}_b^{i+1}. \tag{3-26}$$

The nodes of the ray, however, are constrained to lie on the interfaces, so it is necessary to project these slownesses onto their respective interfaces using the vectors, and map the three-component slownesses into corresponding two-component vectors.

The projection vectors are given by

$$\boldsymbol{u}_b = \begin{bmatrix} 1 \\ 0 \\ \left.\frac{\partial z}{\partial x}\right|_b \end{bmatrix} \text{ and } \boldsymbol{v}_b = \begin{bmatrix} 0 \\ 1 \\ \left.\frac{\partial z}{\partial y}\right|_b \end{bmatrix}, \tag{3-27}$$

where $\left.\frac{\partial z}{\partial x}\right|_b$ is the gradient of the interface at a given node. These gradients are found by a bilinear interpolation of the dip grids, which are precalculated for each interface by central differencing of the depth grid when it is created.

Therefore the two-component slownesses, which give the traveltime change due to a node movement along the interface, can be written as

$$\frac{\partial T}{\partial x'_b} = \nabla_b T \cdot \boldsymbol{u}_b \tag{3-28}$$

and

$$\frac{\partial T}{\partial y'_b} = \nabla_b T \cdot \boldsymbol{v}_b. \tag{3-29}$$

The expressions for the slownesses in equation 3-24 and equation 3-25 can be differentiated with respect to the node positions to give the second derivatives matrix. Again this Hessian matrix can be projected on to the interfaces, and the $3 \times 3$ matrices reduced to $2 \times 2$ blocks. This projection requires an estimate of the curvature of the interface at the node position which is calculated by central differencing of the dip estimates.

### 3.9.2 Anisotropic Extensions

The problem of ray-tracing through general, inhomogenous, anisotropic media was essentially solved by Cerveny (1972). This solution is, however, unnecessarily general and therefore expensive for the needs of the inversion.

Shearer and Chapman (1988) show that the projection of a ray in a factorised anisotropic medium with linear spatial velocity variation on to the plane containing the slowness and velocity gradient vectors is a segment of the curve given by the intersection of the slowness surface and that plane, rotated by $\pi/2$ and scaled. In accordance with Snell's law, which still holds in anisotropic media, this plane is constant for the ray, as is the value of the slowness component normal to the velocity gradient. The "centre" of the ray, i.e. the effective origin of the scaled, rotated slowness surface, is located on the plane defined by the zero-velocity surface. Paul Williamson (unpublished note, 1996) adapted the results of Shearer and Chapman (1988) to calculate the slowness at the end of a given ray segment, as required by the ray-tracing technique presented here. Given these anisotropic slownesses, the solution of the ray-tracing problem proceeds as before.

When considering a TIV medium (see section 4.4.1) with a vertical velocity gradient, the rays stay in the plane defined by the slowness and gradient vectors. When the velocity gradient lies in a direction other than vertical, there is an added complexity. Although the phase-slowness vector remains in a plane with the gradient vector, the group velocity will, in general, have some component out of this plane resulting in some out-of-plane displacement of the ray. A formula for the out-of-plane displacement arising from the divergence of the phase and group directions in anisotropic media is also given by

Shearer and Chapman (1988). Levin (1990) demonstrated that for anisotropic media the application of Snell's law gives a transcendental equation that must be solved iteratively. This is because Snell's law involves both the direction and the velocity of ray, but the velocity is not known at the outset because it is itself a function of the yet-to-be-determined ray direction.

The calculation of the anisotropic Hessian is algebraically daunting, and would produce results considerably more complicated than those for the isotropic case, which is already complicated enough. However, since the ray-tracing is iterative it seems probable that a damped version of the isotropic Hessian, although incorrect, will suffice. The tests that have been carried out confirm this intuitive judgement and show that the ray-tracing still rapidly converges for realistic values of the anisotropy.

# 4.0 The Inverse Problem

## 4.1 Overview

Once the modelling is complete, the consistency of the modelled data with the actual data must be checked and inconsistencies used to update the model. A standard linear inversion technique is presented and applied, although there are complications to the analysis because of the different dimensions of the model parameters. Generally, we aim to invert for the minimum number of model parameters. This makes the model as simple as possible and requires the minimum *a priori* information. This is important as poorly constrained velocity variations are to be avoided. Use of a minimum number of parameters also reduces the non-uniqueness of the result, which is a very important part of the inversion and is studied through construction of a resolution matrix.

## 4.2 Classical Inversion Approach

A general inversion methodology is available whenever the solution to the forward problem is known. If we perturb the model parameters from our initial guess, then the corresponding changes in the modelled data can be calculated. In linear inversion approaches it is assumed that the relationship between this perturbation and the modelled data is linear, and therefore

$$\Delta \boldsymbol{d} = \boldsymbol{J} \Delta \boldsymbol{m}, \tag{4-1}$$

where: $\Delta \boldsymbol{m} = (\Delta m_1, \Delta m_2, \ldots, \Delta m_m)^T$ is an $m$-component vector in the model space containing the perturbations of the model parameters, $\Delta \boldsymbol{d} = (\Delta d_1, \Delta d_2, \ldots, \Delta d_n)^T$ is an $n$-component vector in the data space containing the corresponding perturbations in the modelled data, and $\boldsymbol{J}$ is an $n \times m$ matrix that relates the two.

An objective function, $D$, can be defined which quantitatively measures the misfit between the modelled and the actual data. The problem is to find the set of model parameters $\boldsymbol{m}$ that minimize this function:

$$2D(\boldsymbol{m}) = [\boldsymbol{d}(\boldsymbol{m}) - \boldsymbol{d}_{obs}]^T [\boldsymbol{d}(\boldsymbol{m}) - \boldsymbol{d}_{obs}] \tag{4-2}$$

Here $\boldsymbol{d}(\boldsymbol{m})$ is a column vector with $n$ components containing the modelled data values and $\boldsymbol{d}_{obs}$ is a column vector with $n$ components containing the observed data values.

Any inversion scheme should consist of two parts. Firstly, a particular solution, $\hat{\boldsymbol{m}}$, that minimises $D$ must be found and, secondly, the resolution and error associated with this particular solution must be calculated. The objective function can be minimized using an iterative Gauss-Newton technique as used in the ray-bending algorithm in Section 3.8.1. Differentiating equation 4-2 with respect to the model parameters yields

$$\nabla \boldsymbol{D}(\boldsymbol{m}) = \boldsymbol{J}^T [\boldsymbol{d}(\boldsymbol{m}) - \boldsymbol{d}_{obs}]. \tag{4-3}$$

Another stage of differentiation, ignoring higher order terms which are generally much smaller near the solution, and much harder to calculate, approximates the Hessian matrix

$$\nabla\nabla \boldsymbol{D}(\boldsymbol{m}) \approx \boldsymbol{J}^T \boldsymbol{J}, \tag{4-4}$$

and the model updates required to minimize $D$ are then given by

$$\Delta \boldsymbol{m} = (\nabla\nabla \boldsymbol{D})^{-1} \nabla \boldsymbol{D}, \tag{4-5}$$

which from equation 4-3 and equation 4-4 are approximated as,

$$\Delta \boldsymbol{m} = (\boldsymbol{J}^T \boldsymbol{J})^{-1} \boldsymbol{J}^T \Delta \boldsymbol{d}(\boldsymbol{m}). \tag{4-6}$$

In order to calculate these updates directly, the matrix $\boldsymbol{J}^T \boldsymbol{J}$ must be non-singular, i.e. $m$ of the $n$ equations must be linearly independent. However, if a lack of data means that this is not the case, the inversion will be unstable. In order to study the properties of the linearised inverse problem, a singular-value decomposition approach was used (Jackson, 1972). Following Lanczos (1961) an $n \times m$ matrix $J$ can be factorised as

$$J = U \Lambda V^T , \tag{4-7}$$

where $U$ is a $n \times n$ matrix of orthogonal singularvectors that span the data space, $V$ is a $m \times m$ matrix of orthogonal singularvectors that span the model space, and $\Lambda$ is a semi-diagonal $n \times m$ matrix that contains the so-called singular values. Equation 4-7 is known as a singular-value decomposition.

The matrix $U$ can be separated into $U_p$ and $U_0$, where $U_p$ contains the singularvectors with non-zero singular values and $U_0$ contains the singularvectors with zero singular values. Similarly, $V$ can be divided into $V_0$ and $V_p$, and the matrix $\Lambda$ can be partitioned as

$$\Lambda = \begin{bmatrix} \Lambda_p & 0 \\ 0 & 0 \end{bmatrix} , \tag{4-8}$$

where $\Lambda_p$ is a $p \times p$ diagonal matrix that contains the non-zero singular values of $J$ arranged in decreasing size. $U_0$ and $V_0$ are referred to as ‘null-spaces’. The matrix $J$ can be re-expressed as

$$J = U_p \Lambda_p V_p^T . \tag{4-9}$$

This factorisation is very interesting as it shows that $J$ can be constructed from $U_p$ and $V_p$ alone. This demonstrates that the linear combinations of data values contained in $U_0$ are completely independent of the model parameters, and that the linear combinations of model parameters contained in $V_0$ are totally unresolved by the available data. This means that when a singular value is zero, the corresponding singular vector in data space cannot be mapped into model space or vice versa. Data vectors or model vectors with zero singular values belong to the *null space* and cannot be resolved. When a singular value is not zero, but is small compared with the largest one (i.e. the condition number is large), the contribution of the corresponding eigenvectors to the solution must be eliminated or attenuated, that is regularised, because the matrix inversion may become unstable.

The use of singular value decomposition to solve inverse problems has been thoroughly discussed in the literature (e.g. Aki and Richards, 1980). This analysis can be used to check whether the inversion of the matrix $J^T J$ is stable. From equation 4-9 it is clear that

$$J^T J = V_p \Lambda_p U_p^T \cdot U_p \Lambda_p V_p^T \tag{4-10}$$

and, because of orthogonality of the eigenvectors in the data- and model-space, we have $U_p^T U_p = V_p^T V_p = I$ and therefore

$$J^T J = V_p \Lambda_p^2 V_p^T . \tag{4-11}$$

The orthogonality also means that the inverse is given by

$$(J^T J)^{-1} = V_p^T \Lambda_p^{-2} V_p . \tag{4-12}$$

Using this approach to solve equation 4-6 is known as the generalised inverse. In practice, however, there are two main problems. Firstly, if some of the eigenvalues are small but non-zero, errors in the data could cause strong fluctuations in the solution. One way of suppressing these undesirable effects is to use the damped least-squares approach

(Levenberg, 1944), which consists in adding a positive constant to the main diagonal of the matrix $J^T J$ so that the solution is modified to

$$\Delta m = (J^T J + \alpha I)^{-1} J^T \Delta d(m). \tag{4-13}$$

The damping factor $\alpha$ ensures the non-singularity of the matrix and also stabilises the inversion process.

The second problem is that model updates calculated using this scheme can be unphysical if the 'null-space' corresponds to parameter combinations without clear physical meaning. This means that a generalised inversion approach is often unsatisfactory.

## 4.3 Use of *A Priori* Information to Constrain the Inversion

Another, more physical, way of removing the non-uniqueness of a particular inverse problem is to add constraints that determine the type of model being sought. Tarantola and Valette (1982) proposed a stochastic inverse that explicitly includes *a priori* information in the inverse problem through covariance matrices $C_d$ and $C_m$ of the data and parameters, respectively. The objective function is written

$$2D(m) = [d(m) - d_{obs}]^T C_d^{-1} [d(m) - d_{obs}] + [m - m_0]^T C_m^{-1} [m - m_0], \tag{4-14}$$

where $m_0$ is the *a priori* solution of the problem. Again, the first term measures how well the modelled data matches the real data, and the second term is the distance from the *a priori* set of model parameters and helps to overcome the non-uniqueness of the inversion. The observed data are usually assumed to be mutually independent and therefore the data covariances can be represented by a diagonal matrix, with values determined by the standard deviations, $\sigma_{i=1,n}$, a quantitative measure of the accuracy of each piece of data:

$$C_d = \begin{bmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & \sigma_n^2 \end{bmatrix}. \tag{4-15}$$

The *a priori* model parameter covariance matrix defines the expected deviation of the model parameters around the *a priori* model. So, for example, if a given model parameter is well estimated prior to the inversion, this information can be included by specifying a small deviation in the model parameter covariance matrix. Again, this matrix is almost diagonal, although some non-diagonal terms may be included to introduce constraints that involve linear combinations of parameters. For example, non-diagonal terms may be used to constrain the ratio of two parameters, e.g

$$\begin{bmatrix} m_1 & m_2 \end{bmatrix} \begin{bmatrix} 1 & -n \\ -n & n^2 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} = (m_1 - nm_2)^2. \tag{4-16}$$

These *a priori* covariance matrices are not very easy to define and are, of course, heavily model-dependent. One independent source of constraint comes from the well logs. For example, the vertical gradient, $k_z$, is often poorly constrained by the seismic and so the sonic log can be upscaled to estimate $k_z$ at the wells. Similarly, the petrophysical modelling scheme of Xu and White (1996) may be used to construct an anisotropic earth model at the log scale from a standard suite of logs. These predicted anisotropy parameters can be upscaled to give parameters for the seismic units (Williamson et al., 1997). Despite the fact that the well log data describes the region only in the immediate vicinity of the well, it can still be used as an *a priori* constraint and thus integrate all available data in a rigorous way.

As before, equation 4-14 can be differentiated with respect to the model parameters to give the relations

$$\nabla D(\boldsymbol{m}) = \boldsymbol{J}^T C_d^{-1}[\boldsymbol{d}(\boldsymbol{m}) - \boldsymbol{d}_{obs}] + C_m^{-1}[\boldsymbol{m} - \boldsymbol{m}_0] \tag{4-17}$$

and, ignoring higher order terms as before,

$$\nabla\nabla D(\boldsymbol{m}) \approx \boldsymbol{J}^T C_d^{-1} \boldsymbol{J} + C_m^{-1}. \tag{4-18}$$

The model updates can then be calculated using equation 4-5. Iterative application of these updates, $\Delta\boldsymbol{m}$, give us a particular solution, $\hat{\boldsymbol{m}}$.

$$\Delta\boldsymbol{m} = [\boldsymbol{J}^T C_d^{-1} \boldsymbol{J} + C_m^{-1}]^{-1} (\boldsymbol{J}^T C_d^{-1}[\boldsymbol{d}(\boldsymbol{m}) - \boldsymbol{d}_{obs}] + C_m^{-1}[\boldsymbol{m} - \boldsymbol{m}_0]) \tag{4-19}$$

## 4.4 Model Parameterisation

A very important part of any inversion scheme is the choice of model parameterisation that is being inverted for, and this should be considered as a strong *a priori* constraint. The method presented here is considerably different from the majority of tomographic inversion schemes in that the model geometry is not parameterised and inverted for. Instead, it is uniquely defined by the zero-offset times and dips, given by the demigration of the time-migrated interpretation, and the current estimate of the velocity parameters. We are, therefore, only inverting for the velocity parameters.

This approach has the advantage that the number of inverted parameters is reduced. Also, as the depth surface is not defined parameterically, it is easier to include discontinuities caused by faults which cannot easily be handled by spline surfaces or other parameteric forms. The disadvantage is that the demigrated zero-offset representation of the layers is assumed to be perfect and is invariant throughout the inversion process. In practice, however, this assumption is not very limiting because it is usually valid in areas with moderate enough structure for a post-stack approach to be successfully applied. Also, if the data were incorrectly interpreted, they can always be iteratively reinterpreted, either in time or depth, after improving the migration velocity model with the inversion result.

A layer-based parameterisation is used to define the model in the inversion as shown in figure 3-19. This means that the velocity field is defined in a geologically meaningful way and gives a good compromise between spatial variability and a restricted number of model parameters. The boundaries between layers, defined in depth, are stored on regular grids and are calculated, given the current estimate of the model parameters, using the depth-migration procedure described in section 2.8.1.

### 4.4.1 Anisotropy

One of the main objectives of this work was to produce a single model that was consistent with both the surface seismic and the well data. As described in Chapter 1, this required the introduction of anisotropy. Before discussing the exact velocity parameterisation it is, therefore, useful to give a very brief overview of anisotropic wave propagation.

Anisotropy is the variation of propagation velocity with direction. It may be caused by a preferred orientation of anisotropic mineral grains (such as in a massive shale formation), preferred orientation of intrinsically isotropic minerals (e.g. flat-lying clay platelets), preferred orientation of cracks, or thin bedding of isotropic or anisotropic layers (Thomsen, 1986). In linearly elastic material each component of stress is linearly dependent upon every component of strain (Nye, 1957). This dependency is characterised by a fourth order tensor, $C_{ijkl}$; however, because of fundamental symmetry properties of the stress field there are at most 21 independent components. When combined with the effects of heterogeneity, such a multitude of parameters will probably always be unresolvable from a seismic experiment and so certain symmetries of the elastic properties are assumed. The simplest realistic form of anisotropy is transverse isotropy (TI), or hexagonal symmetry. Transverse isotropy has a single distinct direction, usually taken to be the vertical (TIV medium) with the other two directions being symmetrical. TI media require five independent components to describe the elastic modulus tensor, and solution of the wave equation yields three independent solutions polarised in mutually orthogonal directions. Thomsen (1986), in a ground-breaking paper on anisotropy, proposed a set of five such parameters:

$$V_0 = (c_{3333}/\rho)^{1/2} \tag{4-20}$$

$$\beta_0 = (c_{2323}/\rho)^{1/2} \tag{4-21}$$

$$\varepsilon = \frac{c_{1111} - c_{3333}}{2c_{3333}} \tag{4-22}$$

$$\delta = \frac{(c_{1133} + c_{2323})^2 - (c_{3333} - c_{2323})^2}{2c_{3333}(c_{33} - c_{2323})} \tag{4-23}$$

$$\gamma = \frac{c_{1212} - c_{2323}}{2c_{2323}}. \tag{4-24}$$

He demonstrated that by considering the anisotropy to be weak (<20%), but generally within the realms of seismic exploration, the equations simplified considerably and presented relatively simple first-order expressions for the phase velocities of the three wavetypes.

In general, the P-wave phase velocity in a weakly TIV medium is given by (Thomsen,1986) as:

$$v(\theta) \approx v_0(1 + \delta \sin^2\theta \cos^2\theta + \varepsilon \sin^4\theta) \ , \tag{4-25}$$

where $\theta$ is the phase angle measured from the vertical. From equation 4-25 it is clear that $\varepsilon$ controls the horizontal velocity and that $\delta$ is most influential at around $45°$.

The S-wave phase velocities are

$$v_{SV}(\theta) = \beta_0\left(1 + \frac{\alpha_0^2}{\beta_0^2}(\varepsilon - \delta)\sin^2\theta\cos^2\theta\right) \tag{4-26}$$

and

$$v_{SH}(\theta) = \beta_0(1 + \gamma\sin^2\theta). \tag{4-27}$$

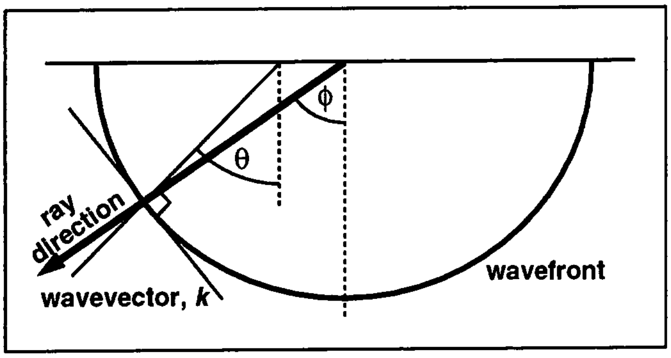


*Figure 4-1: Definition of the phase angle,* $\theta$ *and the group angle,* $\phi$.

An interesting effect of anisotropic wave propagation is shown in figure 4-1. It is clear that the ray, or group angle, $\phi$, which defines the angle of energy propagation, differs from the phase angle, $\theta$, which defines the local direction of the wavefront propagation. The group, or ray, velocity as a function of the phase angle (Berryman, 1979) is;

$$V(\varphi(\theta))^2 = \left[v(\theta)^2 + \left(\frac{\partial v}{\partial\theta}\right)^2\right]. \tag{4-28}$$

This means that for anisotropic media the zero-offset rays are not normal to the reflecting interface. For isotropic media the wavefront is spherical and the phase and group velocities are identical.

Traditionally, seismic processing has assumed that wave propagation through the earth has been isotropic, despite the fact that most crustal rocks have been demonstrated through laboratory measurements to be weakly anisotropic. Kaarsberg (1959) showed from laboratory measurements that the P-wave velocity was 10-20% higher in the direction parallel to the bedding plane than the velocity normal to the bedding plane. The reason for this apparent dichotomy between industrial practice and physical reality is mainly due to the fact that the most commonly occurring type of anisotropy, transverse isotropy, is not usually detectable from surface seismic data alone, with the angular dependence of the velocities disguised in the uncertainties to the depth of the reflector (Thomsen, 1986). This is primarily due to the limited range of propagation angles found in most surface seismic data, which allows more pragmatic approaches, such as scaling of the interval velocity function to tie the wells, to work well enough for conventional interpretation. There may also be other factors that mask the effects of the anisotropy. Lynn et al. (1991) suggest that, in many cases, the effects of anisotropy roughly cancel those of heterogeneity. Nowadays, however, the literature is full of examples where taking anisotropy into account during seismic processing is demonstrated to be beneficial (e.g. Ball, 1995).

### 4.4.2 Velocity Parameterisation

The velocity within each layer is considered to be transversely isotropic with a vertical symmetry axis (TIV). The inversion is primarily concerned with P-wave propagation; however, some of the well seismic events can be associated with mode-conversions and then S-wave propagation must also be considered. Within each layer, the instantaneous velocity field, as a function of the position, $\boldsymbol{r} = (x, y, z)$, and the phase angle ,$\theta$, is parameterised as:

$$v(\boldsymbol{r},\theta) = (v_0 + \boldsymbol{k}\cdot\boldsymbol{r} + v(x,y))(1 + \delta\sin^2\theta\cos^2\theta + \varepsilon\sin^4\theta), \qquad (4\text{-}29)$$

where $v_0$ is the vertical P-wave velocity at the origin, $\boldsymbol{k} = \left(\frac{\partial v}{\partial x},\frac{\partial v}{\partial y},\frac{\partial v}{\partial z}\right)$ is a three-component velocity gradient, and $\delta,\varepsilon$ are Thomsen's parameters. The term $v(x,y)$ defines higher-order lateral variation in the velocity field and its form can be chosen with considerable flexibility. Currently it is expressed as a polynomial:

$$v(x,y) = \sum_{j=2}^{N}\left(\sum_{i=0}^{j} k_{ij}x^{j-i}y^{i}\right), \qquad (4\text{-}30)$$

where $N$ is the order of the polynomial expansion. In the inversion these polynomials are mapped into Legendre polynomials, which have the advantage of being orthogonal (i.e. the integral of the product of any pair of polynomial terms over the region is zero), and are hence independent of each other. The inverted Legendre polynomial coefficients are then mapped back into the $k_{ij}$ coefficients of equation 4-30. This stablises the inversion and reduces numerical problems during computation.

## 4.5 Uncertainty Estimation on the Inverted Result

An important topic is the estimation of uncertainties on the inversion result. This is a very complex problem that is extremely difficult to solve rigorously due to the large number of potential sources of error and their complex interaction with each other. Broadly speaking, however, the uncertainty on the result has two main components. The first is related to the input data, and the second to the methodology itself. We can go some way towards addressing the impact of errors in the input data, as discussed in the following sections. Methodology errors, however, are harder to quantify and this problem remains largely unsolved. Ultimately, from a practical point of view we are not interested in the uncertainties on the inverted model parameters, but on the final depth maps. The uncertainty on the depth maps can then be mapped into volumetric uncertainties in a later stage.

## 4.6 Resolution Matrix

The first source of uncertainty related to the input data is the fact that the data may be insufficient to uniquely resolve the model parameters. This means that there are many combinations of model parameters that minimize the objective function. Each solution, however, may result in significantly different depth maps and volumetric estimations. Backus and Gilbert (1968) introduced the concept of the resolution kernel as a measure of non-uniqueness. They demonstrated that one way of measuring the resolution is to consider perturbing the model away from the solution and then seeing how well the inversion recovers the perturbation. This is the basis for the resolution matrix.

Consider a perturbation of $\Delta\hat{\boldsymbol{m}}$ away from the true solution, $\hat{\boldsymbol{m}}$. The corresponding change in the observed data, $\Delta d_{obs}$, can be calculated from equation 4-1:

$$\Delta\boldsymbol{d}_{obs} = \boldsymbol{J}\Delta\hat{\boldsymbol{m}} \tag{4-31}$$

The effect of changing the observed data on the model parameters recovered after inversion, $\Delta\boldsymbol{m}_r$, is found by differentiation of equation 4-20:

$$\Delta\boldsymbol{m}_r = -[\boldsymbol{J}^T\boldsymbol{C}_d^{-1}\boldsymbol{J} + \boldsymbol{C}_m^{-1}]^{-1}\boldsymbol{J}^T\boldsymbol{C}_d^{-1}\Delta\boldsymbol{d}_{obs}. \tag{4-32}$$

Therefore, if we consider a perturbation in model parameters, $\Delta\hat{\boldsymbol{m}}$, away from the solution, $\hat{\boldsymbol{m}}$, then from equation 4-32 and equation 4-31;

$$\Delta\boldsymbol{m}_r = -[\boldsymbol{J}^T\boldsymbol{C}_d^{-1}\boldsymbol{J} + \boldsymbol{C}_m^{-1}]^{-1}[\boldsymbol{J}^T\boldsymbol{C}_d^{-1}\boldsymbol{J}]\Delta\hat{\boldsymbol{m}} \tag{4-33}$$

$$\Delta\boldsymbol{m}_r = \boldsymbol{A}\Delta\hat{\boldsymbol{m}}. \tag{4-34}$$

This formulates the recovered perturbation, $\Delta \boldsymbol{m}_r$, as a weighted average of the true perturbation with the weights given by the row vectors of $\boldsymbol{A}$. This weight matrix, which is the product of the inverse of the regularised Hessian and the unregularised Hessian, is called the resolution matrix. The resolution matrix gives a measure of how badly underdetermined the set of equations is. If it equals the identity matrix, $\boldsymbol{I}$, resolution is perfect and the particular solution is equal to the true solution. The model parameters are shown to be uniquely resolved or resolved only in combination with other parameters. It is important to realise that, although the kinematic data may not be able to define the parameters uniquely, the different solutions may still be differentiable in terms of "image quality" during a subsequent depth-migration of the seismic data (especially when pre-stack). This remains an area for future research.

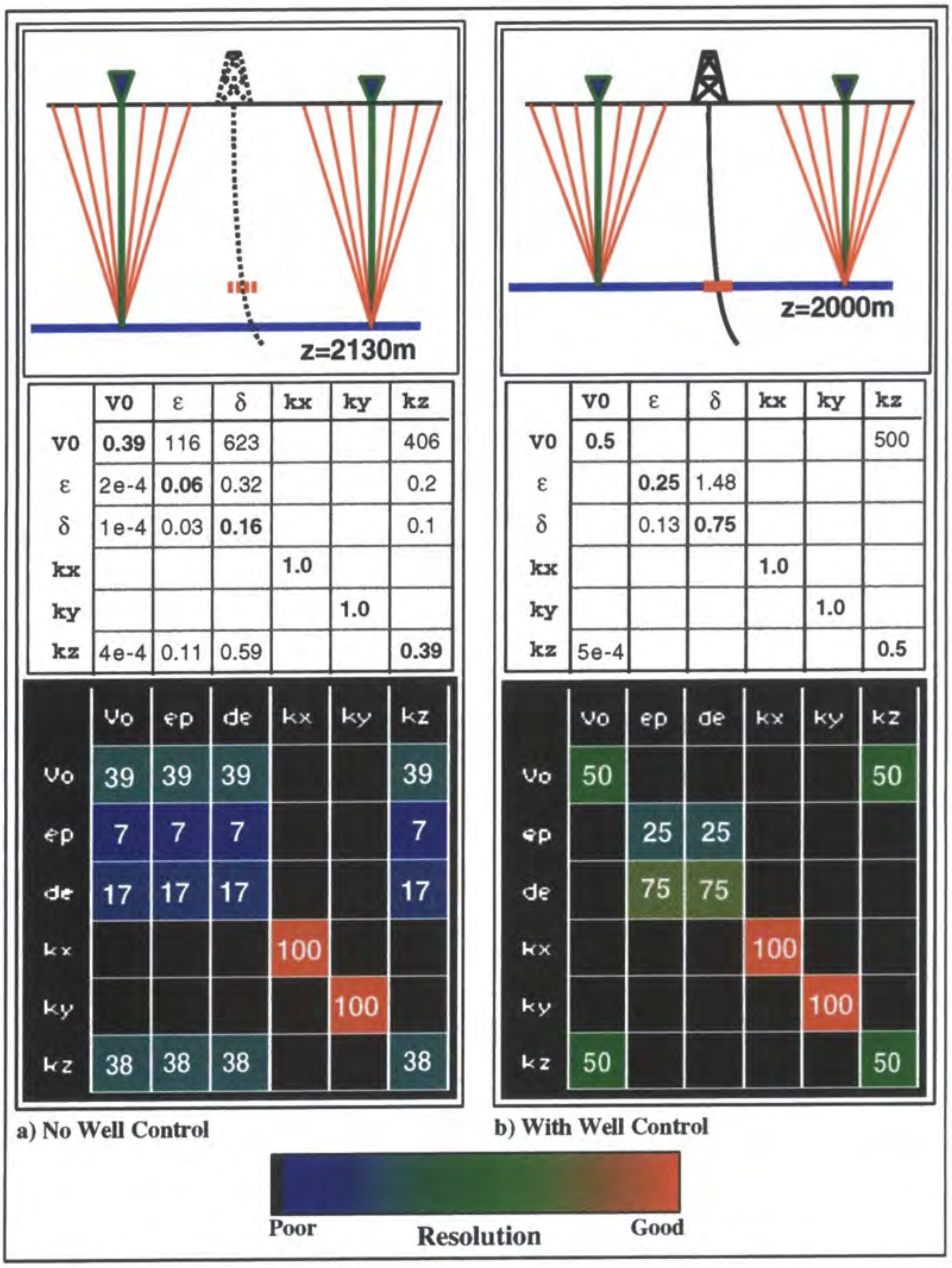
To demonstrate the use of the resolution matrix, I considered the inversion of a synthetic flat layer at a depth of 2000 metres with an anisotropic velocity overburden parameterised by $V0$=2000 m/s, $\delta = 0.05$, and $\varepsilon = 0.15$. A synthetic stacking velocity field was modelled at 16 locations uniformly distributed around the area and a synthetic well marker created. Figure 4-2 shows the results of inversions of these data without well control (left panel) and with well control (right panel).

With reference to equation 4-34, each vertical vector of the resolution matrix (shown in the centre of each panel) corresponds to a given model parameter, and contains the predicted model update after a unity change of that parameter away from the solution, $\hat{\boldsymbol{m}}$. For example, from the left panel of figure 4-2 we can see that if the vertical velocity is perturbed from the solution by 1.0 m/s, the resolution matrix tells us that we will recover only 0.39 m/s back in terms of the vertical velocity, and will change $\varepsilon$ and $\delta$ by 0.0002 and 0.0001 and the vertical gradient by 0.0004. The lateral gradients will remain unchanged. Similarly, if we make a perturbation of 1.0 $s^{-1}$ in one of the lateral gradients, $k_x$, or $k_y$, the inversion recovers the perturbation exactly.

Due to the different dimensions of the model parameters these vectors are difficult to interpret. In order to make the result more presentable, they can be scaled by the Hessian matrix to normalise the changes in terms of the corresponding change in the objective function gradient. The resultant matrix can then be thought of in terms of the percentage of the parameter that is recovered. The rescaled matrices are displayed at the bottom of figure 4-2.

Without well control, only three parameters can be determined. The lateral gradients, $k_x$ and $k_y$, are well resolved, but only a single other parameter which is a linear combination of the other parameters can be found. In this case, there is no evidence for anisotropy from the data and the stacking velocity field is not sufficient to recover the correct layer depth. The simplest velocity model that fits the data is isotropic with no vertical gradient. This gives a layer depth of 2130 metres which is some 6.5% too deep.

When the well is added, the isotropic and anisotropic components separate out, and we observe a $V_0 - k_z$ trade-off and an $\varepsilon - \delta$ trade-off. This result agrees with intuition because, in this case, the layer depth depends only on the vertical velocity and is unaffected by a change in the anisotropy parameters. The addition of the well marker depth, therefore, removes the trade-off between the isotropic and anisotropic parameters. When trade-offs between parameters exist, the final solution will depend strongly on the *a priori* model and *a priori* model covariance matrix, $C_m$, as well as the available data.



|  | V0 | ε | δ | kx | ky | kz |
|---|---|---|---|---|---|---|
| V0 | **0.39** | 116 | 623 |  |  | 406 |
| ε | 2e-4 | **0.06** | 0.32 |  |  | 0.2 |
| δ | 1e-4 | 0.03 | **0.16** |  |  | 0.1 |
| kx |  |  |  | **1.0** |  |  |
| ky |  |  |  |  | **1.0** |  |
| kz | 4e-4 | 0.11 | 0.59 |  |  | **0.39** |

|  | Vo | ep | de | kx | ky | kz |
|---|---|---|---|---|---|---|
| Vo | 39 | 39 | 39 |  |  | 39 |
| ep | 7 | 7 | 7 |  |  | 7 |
| de | 17 | 17 | 17 |  |  | 17 |
| kx |  |  |  | 100 |  |  |
| ky |  |  |  |  | 100 |  |
| kz | 38 | 38 | 38 |  |  | 38 |

**a) No Well Control**

|  | V0 | ε | δ | kx | ky | kz |
|---|---|---|---|---|---|---|
| V0 | **0.5** |  |  |  |  | 500 |
| ε |  | **0.25** | 1.48 |  |  |  |
| δ |  | 0.13 | **0.75** |  |  |  |
| kx |  |  |  | **1.0** |  |  |
| ky |  |  |  |  | **1.0** |  |
| kz | 5e-4 |  |  |  |  | **0.5** |

|  | Vo | ep | de | kx | ky | kz |
|---|---|---|---|---|---|---|
| Vo | 50 |  |  |  |  | 50 |
| ep |  | 25 | 25 |  |  |  |
| de |  | 75 | 75 |  |  |  |
| kx |  |  |  | 100 |  |  |
| ky |  |  |  |  | 100 |  |
| kz | 50 |  |  |  |  | 50 |

**b) With Well Control**

*Figure 4-2: Resolution matrix calculation for a single flat reflector with and without well control*

In this case, the observed $\varepsilon - \delta$ trade-off is due to adding the constraint $(\varepsilon - 2\delta)^2$ to the objective function. This gives recovered perturbation the ratio $\varepsilon = 2\delta$. Without this constraint, the recovered perturbation would depend on the relative sensitivity of the objective function to changes in these parameters. This depends upon the *a priori* deviations on $\varepsilon$ and $\delta$ defined in the *a priori* covariance matrix and, very importantly, on the particular model and data acquisition considered. The stacking velocity field is more sensitive to a change in $\delta$ than $\varepsilon$ and so, without constraint, the majority of the perturbation would come back in $\delta$.

In cases where parameter trade-offs exist, the uncertainty can be assessed by producing several models by moving along the trade-off lines between the different parameters.

## 4.7 *A posteriori* Covariance Matrix

As well as being insufficient, the input data are also uncertain. The inversion result can also help us to assess the effect of this uncertainty of the inverted result by providing the *a posteriori* covariance matrix of the model parameters. This covariance matrix relates error bars in the input data to corresponding error bars in the inverted model parameters.

The *a posteriori* covariance matrix, $C_{\hat{m}}$, is written as;

$$C_{\hat{m}} = \langle \delta m \delta m^T \rangle , \tag{4-35}$$

but from equation 4-19 a perturbation in the data, $\delta d$, can be related to a perturbation $\delta m$ in the model parameters:

$$\delta m = [\nabla\nabla D]^{-1} J^T C_d^{-1} \delta d \quad , \tag{4-36}$$

and therefore

$$C_{\hat{m}} = \langle [\nabla\nabla D]^{-1} J^T C_d^{-1} \delta d \delta d^T C_d^{-1} J ([\nabla\nabla D]^{-1})^T \rangle . \tag{4-37}$$

As $C_d = \langle \delta d \delta d^T \rangle$,

$$C_{\hat{m}} = [\nabla\nabla D] J^T C_d^{-1} J ([\nabla\nabla D]^{-1})^T \tag{4-38}$$

or

$$C_{\hat{m}} = AH^{-1}, \tag{4-39}$$

where $A$ is the resolution matrix defined by equation 4-34.

The diagonal elements of the covariance matrix, $C_{\hat{m}}$, contain the model parameter variances. For example, if we invert only the vertical velocity from the flat layer synthetic, the covariance matrix contains only a single value which is the variance of $V0$. If all the other parameters are set to zero, then the stacking velocity simply equals the vertical velocity. If we assume a standard deviation, $\sigma_{Vs}$, of 50 m/s on the stacking velocity, then the corresponding standard deviation on $V0$ in this simple case is given by,

$$\sigma_{V0} = \left(\sum_{i=1}^{N} \frac{1}{\sigma_{Vs}^{2}}\right)^{-\frac{1}{2}} = \frac{\sigma_{Vs}}{\sqrt{N}}, \tag{4-40}$$

where N is the number of velocity analysis locations used in the inversion. For the synthetic case 16 velocity analysis locations were considered, and therefore $\sigma_{V0} = 12.5$ m/s.

The situation is more complex when we invert for two parameters, for example, $V0$ and $k_z$. The diagonal elements of the resulting covariance matrix contain the variance on the inverted parameters. The off-diagonal terms give the orientation and shape of the contours of the objective function around $\hat{m}$ (equation 4-41):

$$D(m) = D(\hat{m}) + \frac{1}{2}(m - \hat{m})^{T} C_{\hat{m}}^{-1} (m - \hat{m}) \tag{4-41}$$

The covariance matrix can also be used to generate a probability distribution function of the model parameters:

$$P(\boldsymbol{m}-\hat{\boldsymbol{m}}) \propto \exp-\left[\frac{(\boldsymbol{m}-\hat{\boldsymbol{m}})^T C_{\hat{m}}^{-1}(\boldsymbol{m}-\hat{\boldsymbol{m}})}{2}\right]. \qquad (4\text{-}42)$$

Through use of standard multi-variate Monte-Carlo simulations (e.g. Oliver, 1995) many realisations of model parameters, that satisfy the error distribution in the data, can be generated quickly. For each realisation, we can perform a depth migration and calculate the volumetrics from the depth maps. More ambitiously, as we can estimate the movement of the depth surface at any location due to a change in the model parameters, the derivative of the gross rock volume (GRV) with respect to the parameter changes may be estimated directly. Care must be taken, however, because if the perturbation from the solution, $\hat{\boldsymbol{m}}$, is too large then we may move outside the linear regime described by the covariance matrix. This can be checked by modelling and checking the validity of equation 4-41. If there is time to generate enough realisations, the results can be displayed as in figure 4-3 and confidence levels associated with particular reservoir volumes.

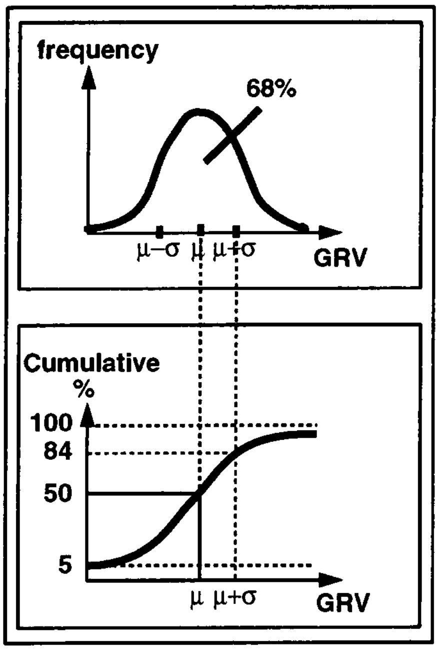


*Figure 4-3: Measuring the uncertainty on the Gross Rock Volume (GRV)*

Another, more pragmatic, way of estimating uncertainties is to actually add errors with some pre-defined distribution to the input data and then reinvert. This type of approach is valid even when the inversion is strongly non-linear, or when the errors in the input data are large enough to move outside the linear regime described by the covariance matrix.

## 4.8 Calculation of the Partial Derivatives

To perform the inversion successfully, we must be able to estimate the partial derivatives of the modelled data with respect to each of the model parameters. Specifically, the modelled data consists of a stacking velocity field, well marker depths and well seismic traveltimes. All of these derivatives are shown to have a very similar form and can be calculated efficiently during the forward modelling. Each calculation relies on calculating how the traveltime along a given ray-path changes due to a perturbation in one of the model parameters.

For ease of calculation, the derivatives are currently calculated using a straight ray approximation, i.e. assuming a locally constant velocity field with constant anisotropy parameters. Figure 4-4 shows an example of a ray, travelling from layer $k-2$ to layer $k$ Initially, we will assume that the overburden velocity field is known and only consider the effects of a change in model parameters in the last interval through which the ray travels. This calculation is appropriate for layer-stripping methodologies. Later, we can generalise the discussion to calculate the traveltime derivatives of a ray with respect to model parameter changes anywhere in the model, as required by the multi-layer inversion scheme.

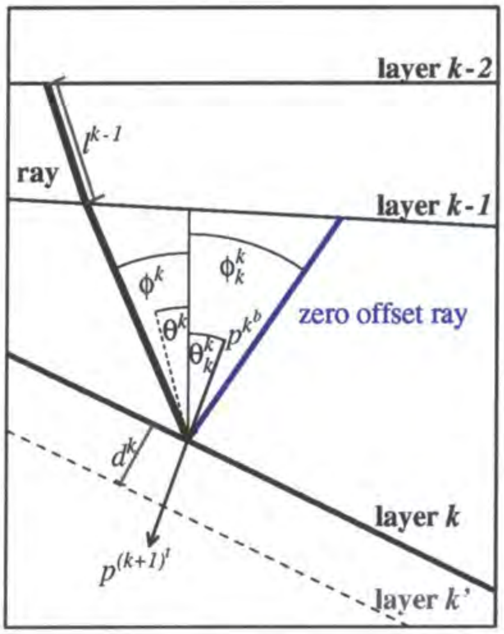


*Figure 4-4: Calculation of the traveltime partial derivatives*

There are two first-order effects that change the traveltime along the ray due to a change in model parameters within the layer $k$ (between layer boundaries $k-1$ and $k$).

- change in traveltime along the unperturbed ray, and
- movement of the interface from $k$ to $k'$, to keep the zero-offset time to layer $k$ constant.

The figure defines a number of parameters that describe the ray-path: the phase and group angles, $\theta, \phi$; the ray-path length, $l$; the component of ray slowness normal to the interface, $p$; and the component of displacement of the interface boundary, $d$, along the direction of the zero-offset ray. We can also define the traveltime, $t$, and the phase and group velocity, $v, V$.

The indexing of these variables is fairly complex and must be explained. The subscript on a variable refers to the ray-path that is being considered. If no subscript is present then the variable relates to the original ray for which we are trying to find the traveltime derivative. If a subscript does exist, then the variable relates to a zero-offset ray (i.e. no slowness component in a direction tangential to the layer interface) fired from the intersection of the original ray and the layer interface indicated by the index. These zero-offset rays are used to calculate how a layer boundary moves due to a change in model parameters.

The superscript on a variable refers to a segment of a ray through the layer indicated by the index, i.e. *k* refers to the interval between *k-1* and *k*. If no superscript exists then the variable describes the whole ray-path. In addition, when describing a slowness, $p$, an additional superscript of *t* or *b* is added to distinguish between the top and the bottom of a layer interval. This indexing may seem over-elaborate at first sight, but the generality is needed when we extend the discussion to a multi-layer inversion.

For both first-order effects we firstly derive the traveltime change due to a change in average phase velocity along the ray, and then analytically relate this phase velocity change to the model parameters.

### 4.8.1 Change in Traveltime along the Unperturbed Ray

As we are dealing with straight rays we need deal only with average group velocity along the ray $V(\varphi)$. The traveltime $t^k$ for a straight segment of length $l^k$ is then simply given by

$$t^k = \frac{l^k}{V^k(\varphi^k)}. \tag{4-43}$$

Therefore, the change in traveltime due to a change in the group velocity is

$$\frac{\partial t^k}{\partial V^k} = -\frac{l^k}{(V^k)^2} = \frac{-t^k}{V^k}, \tag{4-44}$$

and since the group and phase velocities are related by

$$v(\theta) = V(\varphi)\cos(\varphi - \theta), \tag{4-45}$$

the traveltime derivative with respect to the average phase velocity is

$$\frac{\partial t^k}{\partial v^k(\theta^k)} = \frac{-t^k}{v^k(\theta^k)}\cos(\varphi^k - \theta^k). \tag{4-46}$$

Differentiating equation 4-29 with respect to the model parameters yields

$$\frac{\partial v}{\partial V_0} = A(\theta) \tag{4-47}$$

$$\frac{\partial v}{\partial \delta} = v_0 \sin^2\theta\cos^2\theta \tag{4-48}$$

$$\frac{\partial v}{\partial \varepsilon} = v_0 \sin^4\theta, \tag{4-49}$$

where:

$$A(\theta) = (1 + \delta\sin^2\theta\cos^2\theta + \varepsilon\sin^4\theta) \tag{4-50}$$

The traveltime changes, therefore, for the unperturbed ray, due to a change in the model parameters are given by

$$\frac{\partial t^k}{\partial V_0^k} = \frac{-t^k}{V_0^k}\cos(\varphi^k - \theta^k), \tag{4-51}$$

$$\frac{\partial t^k}{\partial \delta^k} = \frac{-t^k V_0^k \cos(\varphi^k - \theta^k)\sin^2\theta^k\cos^2\theta^k}{A^k(\theta^k)}, \tag{4-52}$$

and

$$\frac{\partial t^k}{\partial \varepsilon^k} = \frac{-t^k V_0^k \cos(\varphi^k - \theta^k)\sin^4\theta^k}{A^k(\theta^k)}. \tag{4-53}$$

As straight rays are being assumed, the traveltime change due to a change in velocity gradient can be re-expressed in terms of the change in the average phase velocity, $v$ due to changing the gradient parameters $\boldsymbol{k}$:

$$v(\theta) = (V_0 + \boldsymbol{k} \cdot \boldsymbol{r})A(\theta) \tag{4-54}$$

$$\frac{\partial t}{\partial k_i} = \frac{\partial t}{\partial v} \cdot \frac{\partial v}{\partial k_i} = r_i \frac{\partial t}{\partial v}. \tag{4-55}$$

In a similar way, the traveltime change due to higher order velocity terms can be given as the change the traveltime with respect to $v$ multiplied by the change in $v$ with respect to the higher order parameter. This makes it very easy to introduce higher order velocity terms of many different forms and, as long as the velocity varies slowly enough to keep the rays fairly straight, the derivatives should be accurate enough.

### 4.8.2 Effect of Moving Interface to Keep Stack-Time Constant

The second first-order effect which must be taken into account is due to the fact that all of the models we build must be consistent with the stack time. In the inversion, whenever the model parameters are changed the model is re-depth migrated, ensuring the depth surfaces are always consistent with the interpreted time data. This is a strong constraint and results in the interface moving as a result of interval parameters changes. It means that the traveltime change along a zero-offset ray due to a perturbation in model parameters is identically zero. This term is related to the fixed traveltime and fixed traveltime-dip criteria of Iverson and Gjøystdal (1996).

As a first approximation, we assume that the interface will move along the direction of the zero-offset ray shot from the interface, as shown on figure 4-4. This component of the traveltime change can then be written as the product of the traveltime change due the movement of the interface and the movement of the interface due to the change in model parameters:

$$\frac{\partial t^k}{\partial m_j} = \frac{\partial t^k}{\partial d^k} \cdot \frac{\partial d^k}{\partial m_j}. \tag{4-56}$$

The first of these terms is the normal component of the zero-offset ray slowness, $p^{k^b}$:

$$\frac{\partial t^k}{\partial d^k} = p^{k^b} = \frac{1}{v^k(\theta_k^k)}, \tag{4-57}$$

which is simply the reciprocal of the phase velocity.

The interface change due to a change in the group velocity is

$$\left.\frac{\partial d^k}{\partial V^k}\right|_{\Delta t_k = 0} = t_k^k, \tag{4-58}$$

where $t_k^k$ is the traveltime along the zero-offset ray, shown in blue on figure 4-4, from the ray segments' intersection with the interface boundary. By shooting a zero-offset ray through this last segment and calculating its traveltime, this factor is determined. These results can be used to determine the movement of the interface, $d^k$, due to changing the phase velocity as

$$\frac{\partial d^k}{\partial v^k(\theta_k^k)} = t_k^k \cos(\varphi_k^k - \theta_k^k), \tag{4-59}$$

and combining equation 4-57 and equation 4-59 gives

$$\frac{\partial t^k}{\partial v^k(\theta_k^k)} = p^{k^b} t_k^k \cos(\varphi_k^k - \theta_k^k). \tag{4-60}$$

As previously, this can be expanded by the chain rule, and we get the traveltime change due to the movement of the interface:

$$\frac{\partial t^k}{\partial V_0^k} = p^{k^b} t_k^k \cos(\varphi_k^k - \theta_k^k) A^k(\theta_k^k) \tag{4-61}$$

$$\frac{\partial t^k}{\partial \delta^k} = p^{k^b} t_k^k V_0^k \cos(\varphi_k^k - \theta_k^k)(\sin\theta_k^k \cos\theta_k^k)^2 \tag{4-62}$$

and

$$\frac{\partial t^k}{\partial \varepsilon^k} = p^{k^b} t_k^k V_0^k \cos(\varphi_k^k - \theta_k^k)(\sin\theta_k^k)^4 . \tag{4-63}$$

## 4.9 Stacking Velocities

The change in the stacking velocity, $V_S$ due to a change in the model parameter, $m_j$, may be re-expressed by the chain-rule to give

$$\frac{\partial V_S}{\partial m_j} = \sum_{i=1}^{n} \frac{\partial V_S}{\partial t_i} \frac{\partial t_i}{\partial m_j}, \tag{4-64}$$

where the summation is over the $n$ offsets used in the stacking velocity calculation.

The first term, the change in the stacking velocity due to the change in the traveltime of one of the offset rays, can be approximated by assuming a linear regression, which means that the stacking velocity is an analytical function of the traveltimes, $t_i$ and offsets, $h_i$ within the ray fan:

$$\frac{1}{V_S^2} = \frac{n\sum h_i^2 t_i^2 - \sum h_i^2 \sum t_i^2}{n\sum h_i^4 - (\sum h_i^2)^2} , \tag{4-65}$$

and therefore the derivative is given analytically by

$$\frac{\partial V_S}{\partial t_i} = \frac{-V_S(nh_i^2 t_i - \sum h_j^2 t_i)}{n\sum h_i^2 t_i^2 - \sum h_i^2 \sum t_i^2} . \tag{4-66}$$

## 4.10 Well Marker Depths

Similarly, if the surface is assumed to move along the zero-offset ray due to a change in the model parameters, then the change in depth at the position of the well marker is found by shooting a zero-offset ray from the intersection of the well trajectory and the depth surface and using equation 4-59 which gives the displacement of the surface along the ray direction.

## 4.11 Multi-Layer Inversion

This approach can be extended to a multi-layer case: the principles remain the same but the algebra gets more complicated. Figure 4-5 shows a ray travelling between layer interfaces *k-1* and *k+1*. The problem we must solve is to calculate the traveltime change along a given raypath due to a change of parameters in a given layer, *k*. As in the layer stripping case, the change has two first order components. The first is the traveltime change along the unperturbed ray; the second is the effect of the movement of, not a single layer boundary as before, but all the layer boundaries beneath layer, *k*, in order to keep the zero-offset traveltime to the layer boundaries constant.

A change in the model parameters in layer *k* changes the traveltime, $t^k$, of the ray through layer *k*, but it also perturbs the layer boundaries of layer *k* and layer *k+1* to *k'* and *k+1'* respectively. The total traveltime change can be written as

$$\frac{\partial t}{\partial v^k} = \frac{-t^k}{v^k}\cos(\varphi^k - \theta^k) + \frac{\partial d^k}{\partial v^k}(p^{k^b} - p^{(k+1)^t}) + \frac{\partial d^{k+1}}{\partial v^k} p^{k^b} . \tag{4-67}$$

The first term in equation 4-67 is the traveltime change along the unperturbed ray and is of the same form as equation 4-46. As in the layer stripping case, the derivatives are calculated using a linear ray approximation, and therefore the phase velocity is assumed to be the average velocity over the ray segment.

The second term relates to the movement of the layer *k* boundary. As the ray continues into layer *k+1*, a layer boundary perturbation effects the traveltime in both adjacent ray segments. With reference to equation 4-56, the traveltime change for both segments, is the product of the derivative of the normal displacement, $d^k$, with respect to the phase velocity, $v^k$, and the component of the zero-offset ray slowness normal to the interface. The normal components of the traveltime slownesses are readily calculated as they are the reciprocal of the local phase velocities. The derivative is evaluated using equation 4-58, and simply equals the traveltime of the zero-offset ray shot from the intersection of the ray with the layer *k* boundary up to the layer *k-1* interface. This normal ray is shown on the diagram in blue:

$$\frac{\partial d^k}{\partial v^k} = t_k^k \ . \tag{4-68}$$

The third term in equation 4-67 is very similar to the second, and concerns the traveltime movement due to layer *k+1*. The calculation of $\frac{\partial d^{k+1}}{\partial v^k}$ is more complex, but possible given the constraint that the zero-offset traveltime to the interface boundaries is constant. If we fire a normal ray, shown here in red, from the intersection of the original ray with the layer boundary, *k+1*, up to layer *k-1*, then we know that the interface *k+1* must move so as to leave the traveltime of this ray unchanged. Therefore we can write

$$\frac{\partial t_{(k+1)}}{\partial v^k} = 0 \ , \tag{4-69}$$

which itself has three terms of a similar form to equation 4-67:

$$0 = \frac{-t_{n(k+1)}^k}{v^k} \cos(\varphi_{(k+1)}^k - \theta_{(k+1)}^k) + \frac{\partial d_{(k+1)}^k}{\partial v^k} (p_{(k+1)k}^{k^b} - p_{(k+1)k}^{(k+1)^t}) + \frac{\partial d^{(k+1)}}{\partial v^k} p^{(k+1)^b} \ . \tag{4-70}$$

The calculation of $\frac{\partial d^{k}_{(k+1)}}{\partial v^{k}}$ requires another zero-offset ray to be fired from the intersection of layer $k$ and the zero-offset ray from layer $k+1$. Given this ray-path, shown on the figure in green, the derivative is found as before.

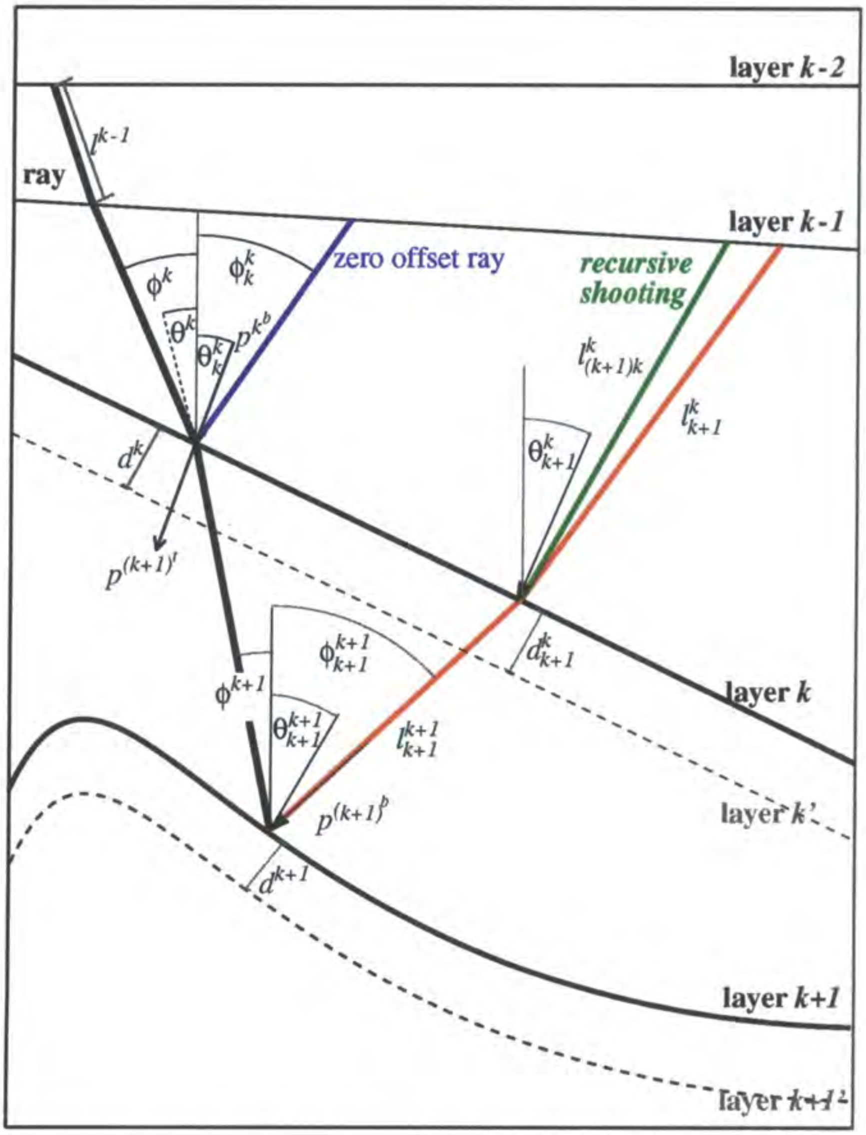


*Figure 4-5: Calculation of the traveltime derivatives for the multi-layer joint inversion*

Rearranging equation 4-70 gives

$$\frac{\partial d^{(k+1)}}{\partial v^{k}} = \frac{1}{p^{(k+1)^{b}}}\left[\frac{t^{k}_{(k+1)}}{v^{k}}\cos(\varphi^{k}_{(k+1)} - \theta^{k}_{(k+1)}) + \frac{\partial d^{k}_{(k+1)}}{\partial v^{k}}(p^{(k+1)^{t}}_{(k+1)k} - p^{k^{b}}_{(k+1)k})\right],$$

(4-71)

and combining equation 4-67 and equation 4-71 gives the required traveltime derivative.

Generally, when there is more than one layer beneath the layer in which the model parameters are being changed, equation 4-67 must be generalised to give

$$\frac{\partial t}{\partial v^{k}} = \frac{-t^{k}}{v^{k}}\cos(\varphi^{k} - \theta^{k}) + \left[\sum_{i=k}^{n-1}\frac{\partial d^{i}}{\partial v^{k}}(p^{(i)^{b}}_{n(i)} - p^{(i+1)^{t}}_{n(i)})\right] + \frac{\partial d^{n}}{\partial v^{j}}p^{(n)^{b}}_{n(n)} = 0, \tag{4-72}$$

where *n* is the index of the last layer through which the ray travels.

Similarly, equation 4-71 becomes

$$\frac{\partial d^{j}}{\partial v^{k}} = \frac{1}{p^{j^{b}}}\left[\frac{t^{k}_{j}}{v^{k}}\cos(\varphi^{k}_{j} - \theta^{k}_{j}) + \sum_{i=k}^{j-1}\frac{\partial d^{i}_{j}}{\partial v^{k}}(p^{(i+1)^{t}}_{ji} - p^{i^{b}}_{ji})\right] \tag{4-73}$$

The terms $\frac{\partial d^{i}_{j}}{\partial v^{k}}$ in equation 4-73 can be found using equation 4-73 itself, and the solution becomes recursive. Practically, this means that zero-offset rays must also be traced from the intersection of all zero-offset rays with the layer boundaries. The recursive nature of the solution means the process could become excessively expensive if the number of layers involved in the derivative calculation becomes too large. In order to optimise the shooting, a new ray is only fired if the angle it makes with the interface is significantly different from that of existing rays.

Figure 4-6 shows the rays that must be fired to calculate the change in traveltime along the ray from the source to receiver given a change in model parameters within any of the layers in the model.

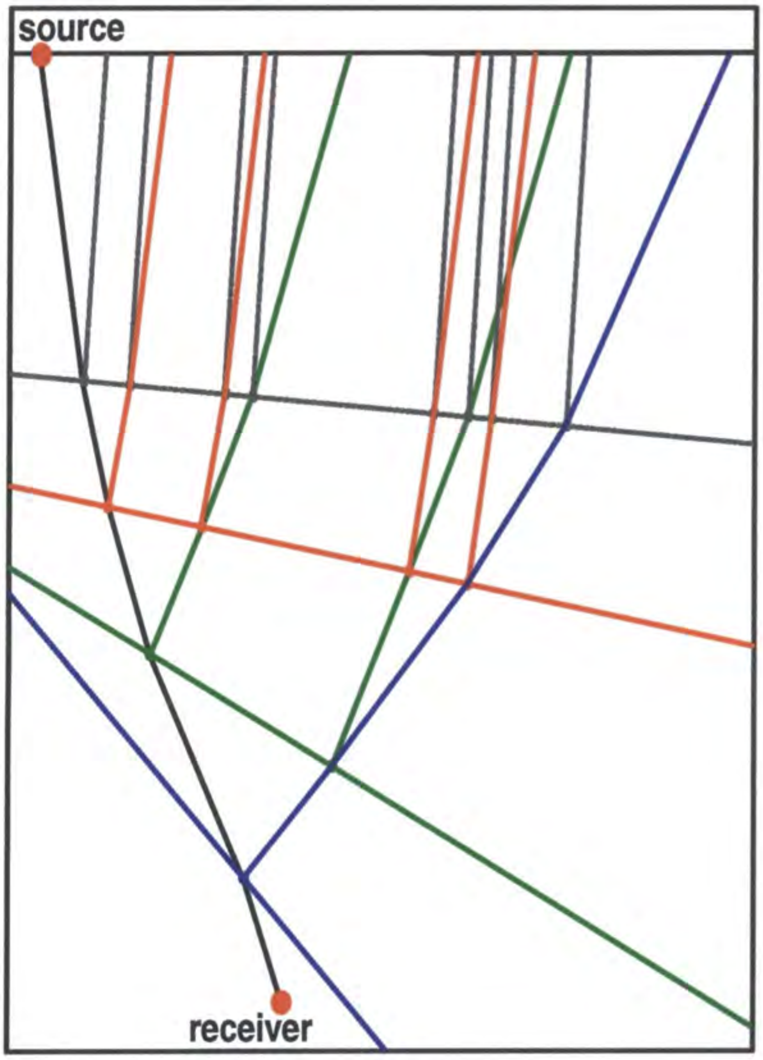


*Figure 4-6: Traveltime derivative calculation using recursive ray-tracing*

# 5.0 Case Studies

## 5.1 Simple Synthetic

### 5.1.1 The Model

A three layer synthetic model was generated in order to test the inversion algorithm (figure 5-1) and assess the impact of the different data types on parameter resolution. The first layer was flat with an isotropic velocity overburden. The second layer was also flat but the velocity was chosen to be anisotropic with a small vertical velocity gradient. The third layer had some mild structure and was strongly anisotropic with a strong vertical velocity gradient. Second-order lateral velocity variations were included in each layer. Figure 5-1 also shows the model parameters as a function of depth at the position of the well.

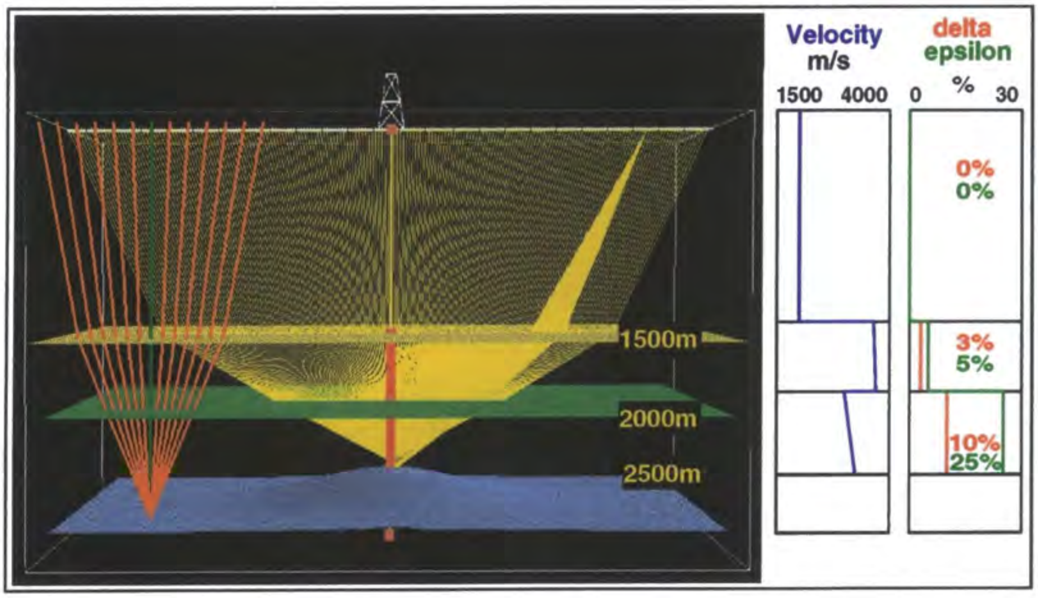


*Figure 5-1: Three layer synthetic model*

The modelled data consisted of: a stacking velocity field, modelled at 81 VA locations spread evenly over the area; well-markers, defined for all three layers; and the modelled traveltimes for a walkaway well seismic acquisition, an offset VSP and a vertical VSP.

Variable amounts of normally distributed noise were added to each of the modelled data to produce a noisy dataset which was inverted in addition to a noise-free dataset.

### 5.1.2 Layer-Strip Results

Initially, a layer-strip inversion was carried out using the stacking velocity field and the well marker depths. Starting from a constant velocity model, each layer required three to four iterations to converge. The results are shown in figure 5-2 for noise-free and noisy datasets. The noisy dataset had Gaussian errors added with standard deviations of 30 m/s for the actual stacking velocity field and 5 m for the well-marker depths. Figure 5-2 also shows the *a posteriori* standard deviations on the inverted anisotropy parameters due this noise estimated from the *a posteriori* covariance matrix. The resolution matrices for the layer-strip inversion are shown in figure 5-3.

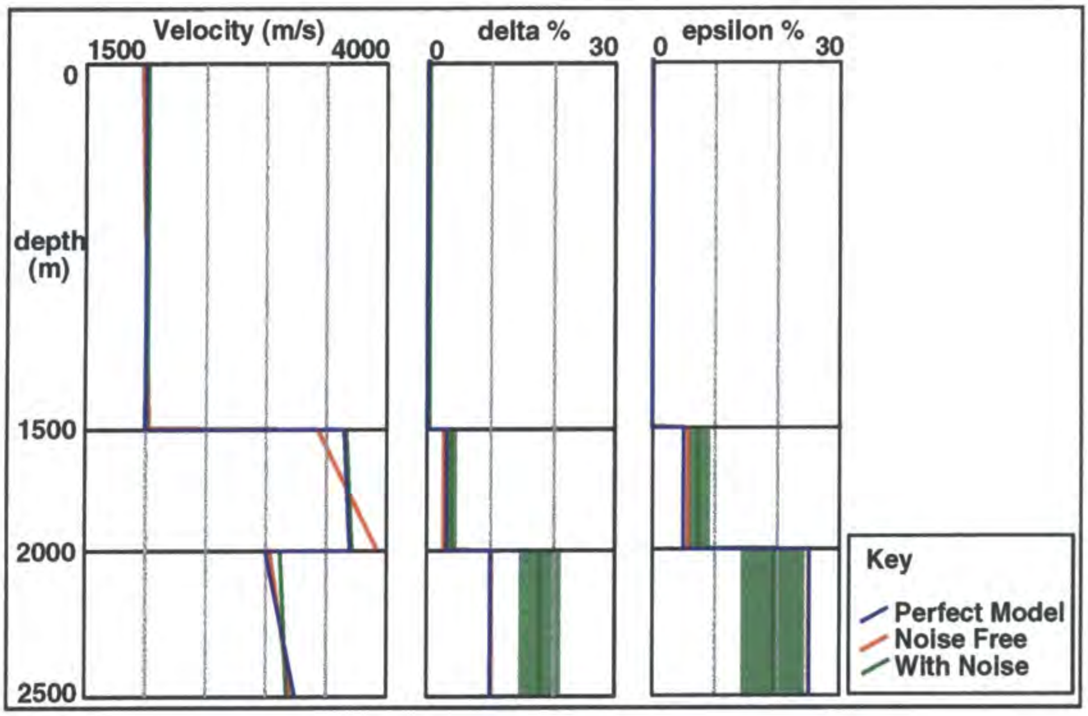
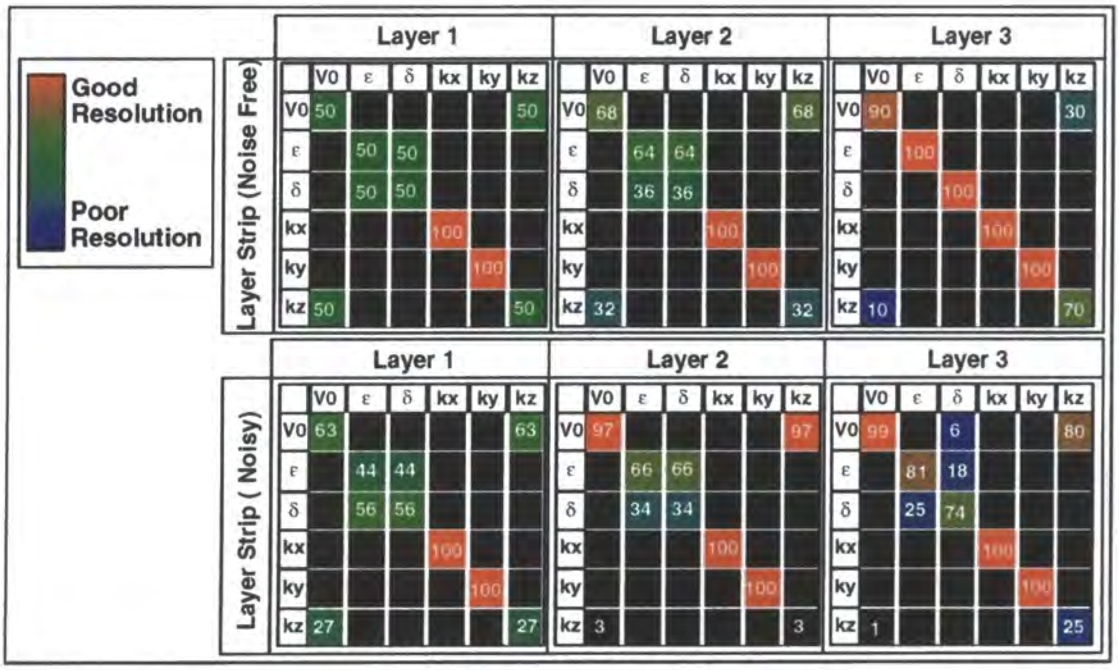


*Figure 5-2: Comparison of the layer-strip inversion results with and without noise*

For the first two flat layers the data are insufficient to resolve both the vertical velocity and the vertical velocity gradient, and can be fitted equally well using a wide variety of

$V0 - k_z$ combinations. The selected solution depends strongly upon the way in which the inversion was constrained, and minimizes the distance from the *a priori* model. Similarly, the anisotropy parameters in the first two layers are not resolved independently. Within the first layer, the inverted velocity model is isotropic because the *a priori* model was isotropic and all the data could be fitted without introducing anisotropy. Within the second layer, the *a priori* model was also isotropic but anisotropy was introduced because it was needed to fit both the stacking velocities and the well marker depths, and the contribution of the data misfit to the objective function was considerably greater than the contribution due to violation of the *a priori* constraints.



Layer Strip (Noise Free) — Layer 1

| | V0 | ε | δ | kx | ky | kz |
|---|---|---|---|---|---|---|
| V0 | 50 | | | | | 50 |
| ε | | 50 | 50 | | | |
| δ | | 50 | 50 | | | |
| kx | | | | 100 | | |
| ky | | | | | 100 | |
| kz | 50 | | | | | 50 |

Layer Strip (Noise Free) — Layer 2

| | V0 | ε | δ | kx | ky | kz |
|---|---|---|---|---|---|---|
| V0 | 68 | | | | | 68 |
| ε | | 64 | 64 | | | |
| δ | | 36 | 36 | | | |
| kx | | | | 100 | | |
| ky | | | | | 100 | |
| kz | 32 | | | | | 32 |

Layer Strip (Noise Free) — Layer 3

| | V0 | ε | δ | kx | ky | kz |
|---|---|---|---|---|---|---|
| V0 | 90 | | | | | 30 |
| ε | | 100 | | | | |
| δ | | | 100 | | | |
| kx | | | | 100 | | |
| ky | | | | | 100 | |
| kz | 10 | | | | | 70 |

Layer Strip ( Noisy) — Layer 1

| | V0 | ε | δ | kx | ky | kz |
|---|---|---|---|---|---|---|
| V0 | 63 | | | | | 63 |
| ε | | 44 | 44 | | | |
| δ | | 56 | 56 | | | |
| kx | | | | 100 | | |
| ky | | | | | 100 | |
| kz | 27 | | | | | 27 |

Layer Strip ( Noisy) — Layer 2

| | V0 | ε | δ | kx | ky | kz |
|---|---|---|---|---|---|---|
| V0 | 97 | | | | | 97 |
| ε | | 66 | 66 | | | |
| δ | | 34 | 34 | | | |
| kx | | | | 100 | | |
| ky | | | | | 100 | |
| kz | 3 | | | | | 3 |

Layer Strip ( Noisy) — Layer 3

| | V0 | ε | δ | kx | ky | kz |
|---|---|---|---|---|---|---|
| V0 | 99 | | 6 | | | 80 |
| ε | | 81 | 18 | | | |
| δ | | 25 | 74 | | | |
| kx | | | | 100 | | |
| ky | | | | | 100 | |
| kz | 1 | | | | | 25 |

*Figure 5-3: Resolution analysis after layer-strip inversion*

In the third layer, the correlation of the stacking velocity field and the structure is sufficient for the vertical gradient and both anisotropy parameters to be fairly well resolved in the case of the noise-free data set. The addition of even a small amount of noise, however, destroys this resolution and *a posteriori* standard deviations on the model parameters are correspondingly high. Errors in the well-marker depths, strongly influence the *a posteriori* parameter deviations because there is no redundancy in these data. The lateral gradients and higher order terms (not shown in the resolution analysis) are well resolved for all layers, even after the addition of noise.

### 5.1.3 Multi-Layer Inversion

Using the result of the layer-strip model as an initial guess, a multi-layer inversion was performed using the well seismic data in addition to the stacking velocities and well marker depths (figure 5-4). It required a further four iterations to minimize the objective function. Errors having a Gaussian distribution with a standard deviation of 2 ms was added to the well seismic traveltimes for the noisy dataset. The resolution matrices for the multi-layer inversion are shown in figure 5-5.

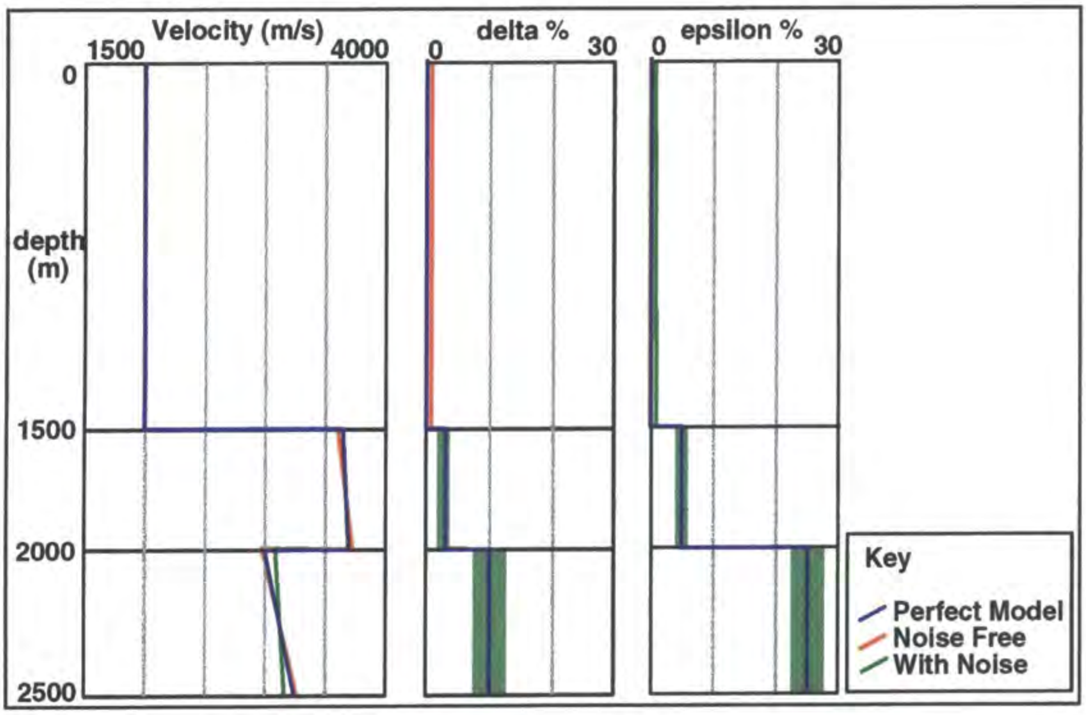
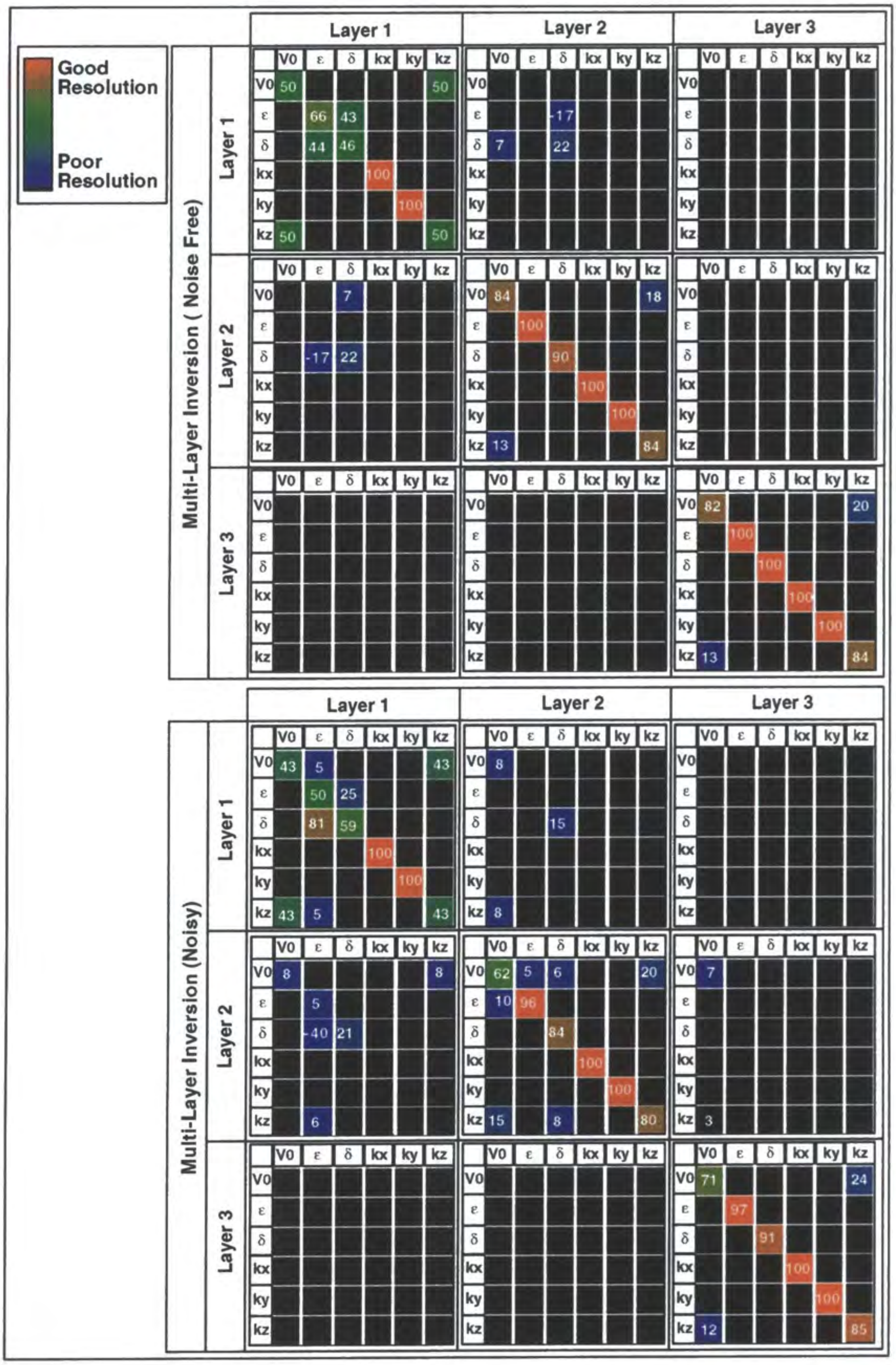


*Figure 5-4: Comparison of multi-layer inversion results with and without noise*

There are no well seismic receivers within the first layer and consequently very little additional information about the model parameters in the first layer. In the second layer, however, the anisotropy parameters are now fairly well resolved, especially epsilon, due to the almost horizontal ray paths present in the well seismic data (figure 5-1). The fact that there are several well seismic receivers within the interval also gives us more information about the $V0 - k_z$ trade-off, as the traveltime difference between receivers contains information about the internal velocity structure of the layer. The resolution matrix also shows trade-offs between parameters in different layers. Interestingly, $\delta$ in the second layer is not resolved independently of the anisotropy parameters in the first layer. Also, in the case of the noisy dataset, the error in the well-marker depth, causes inter-dependency of the vertical velocity and vertical velocity gradient between the first two layers.

Within the last layer, the parameter resolution is almost perfect for the noise-free data and the inverted model is equal to the true model. Also, the resolution is not as badly effected by noise and the *a posteriori* standard deviation on the anisotropy parameters is smaller than after the layer-strip inversion. Again, an error on the well-marker depth introduces small trade-offs in the vertical velocity and vertical velocity gradient in the layer above.

This simple synthetic example shows that the parameter resolution is very dependent upon the structure of the model, the available data and the level of noise. The use of a resolution matrix can be used to quantitatively assess the non-uniqueness of the inversion result for any given model. Generally, as in this case, the addition of well seismic traveltimes is shown to improve parameter resolution.



Multi-Layer Inversion ( Noise Free)

| | | Layer 1 | | | | | | Layer 2 | | | | | | Layer 3 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | V0 | ε | δ | kx | ky | kz | V0 | ε | δ | kx | ky | kz | V0 | ε | δ | kx | ky | kz |
| Layer 1 | V0 | 50 | | | | | 50 | | | | | | | | | | | | |
| | ε | | 66 | 43 | | | | | | -17 | | | | | | | | | |
| | δ | | 44 | 46 | | | | 7 | | 22 | | | | | | | | | |
| | kx | | | | 100 | | | | | | | | | | | | | | |
| | ky | | | | | 100 | | | | | | | | | | | | | |
| | kz | 50 | | | | | 50 | | | | | | | | | | | | |
| Layer 2 | V0 | | | 7 | | | | 84 | | | | | 18 | | | | | | |
| | ε | | | | | | | | 100 | | | | | | | | | | |
| | δ | | -17 | 22 | | | | | | 90 | | | | | | | | | |
| | kx | | | | | | | | | | 100 | | | | | | | | |
| | ky | | | | | | | | | | | 100 | | | | | | | |
| | kz | | | | | | | 13 | | | | | 84 | | | | | | |
| Layer 3 | V0 | | | | | | | | | | | | | 82 | | | | | 20 |
| | ε | | | | | | | | | | | | | | 100 | | | | |
| | δ | | | | | | | | | | | | | | | 100 | | | |
| | kx | | | | | | | | | | | | | | | | 100 | | |
| | ky | | | | | | | | | | | | | | | | | 100 | |
| | kz | | | | | | | | | | | | | 13 | | | | | 84 |

Multi-Layer Inversion (Noisy)

| | | Layer 1 | | | | | | Layer 2 | | | | | | Layer 3 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | V0 | ε | δ | kx | ky | kz | V0 | ε | δ | kx | ky | kz | V0 | ε | δ | kx | ky | kz |
| Layer 1 | V0 | 43 | 5 | | | | 43 | 8 | | | | | | | | | | | |
| | ε | | 50 | 25 | | | | | | | | | | | | | | | |
| | δ | | 81 | 59 | | | | | | 15 | | | | | | | | | |
| | kx | | | | 100 | | | | | | | | | | | | | | |
| | ky | | | | | 100 | | | | | | | | | | | | | |
| | kz | 43 | 5 | | | | 43 | 8 | | | | | | | | | | | |
| Layer 2 | V0 | 8 | | | | | 8 | 62 | 5 | 6 | | | 20 | 7 | | | | | |
| | ε | | 5 | | | | | 10 | 96 | | | | | | | | | | |
| | δ | | -40 | 21 | | | | | | 84 | | | | | | | | | |
| | kx | | | | | | | | | | 100 | | | | | | | | |
| | ky | | | | | | | | | | | 100 | | | | | | | |
| | kz | | 6 | | | | | 15 | | 8 | | | 80 | 3 | | | | | |
| Layer 3 | V0 | | | | | | | | | | | | | 71 | | | | | 24 |
| | ε | | | | | | | | | | | | | | 97 | | | | |
| | δ | | | | | | | | | | | | | | | 91 | | | |
| | kx | | | | | | | | | | | | | | | | 100 | | |
| | ky | | | | | | | | | | | | | | | | | 100 | |
| | kz | | | | | | | | | | | | | 12 | | | | | 85 |

*Figure 5-5: Resolution analysis after multi-layer inversion*

## 5.2 The Elgin / Franklin Fields

The remainder of this chapter will look at the inversion of the Elgin / Franklin field, a real 3D case. Elgin / Franklin is a deep, Upper Jurassic, high-temperature, high pressure, gas condensate accumulation located within the U.K. Central Graben of the North Sea (blocks 22/30c and 29/5b). In this case study, a large area of $13 \times 25\,\mathrm{km}^2$ is considered (figure 5-6). Six seismic horizons were interpreted from the time-migrated block: Top Balder, Top Maureen, Top Hod, Top Hidra, Base Cretaceous Unconformity (BCU) and Top Pentland. The data quality for the shallow horizons allowed automatic 3D horizon picking followed by manual checking. The BCU and Pentland required manual picking and were also heavily faulted (figure 5-6 shows the fault polygons picked on the Pentland layer). There was a problem with multiples generated from the Top Hidra and BCU events which caused the data quality to deteriorate below the BCU. Each time-migrated interpretation was sampled onto a regular grid with a spacing of $50 \times 50$ m.

The time-migration velocity field was based upon a smoothed version of the stacking velocity field modified by a time-varying multiplier, selected after a series of trials, to improve the quality of the image. This field, specified as a three-dimensional grid, was sampled on to the time-migrated interpretations to produce horizon consistent maps that could be used for the demigration. A Kirchhoff demigration operator was applied, despite the fact that finite-difference time-migration was used. This approximation was deemed acceptable, as described in section 2.7.2, because the lateral variation in the time-migration velocity model was not strong.

Stacking velocity analysis was carried out approximately every 500 m, at the locations shown as dots in figure 5-6. In this case, both DMO and a constant velocity pre-stack time-migration (2000 m/s) were applied before stacking velocity analysis. This simple time-migration velocity field should not be confused with the final time-migration velocity field used to migrate the seismic data prior to interpretation. A time-varying mute was also applied which gave 4700 m of offset for the deepest horizons.

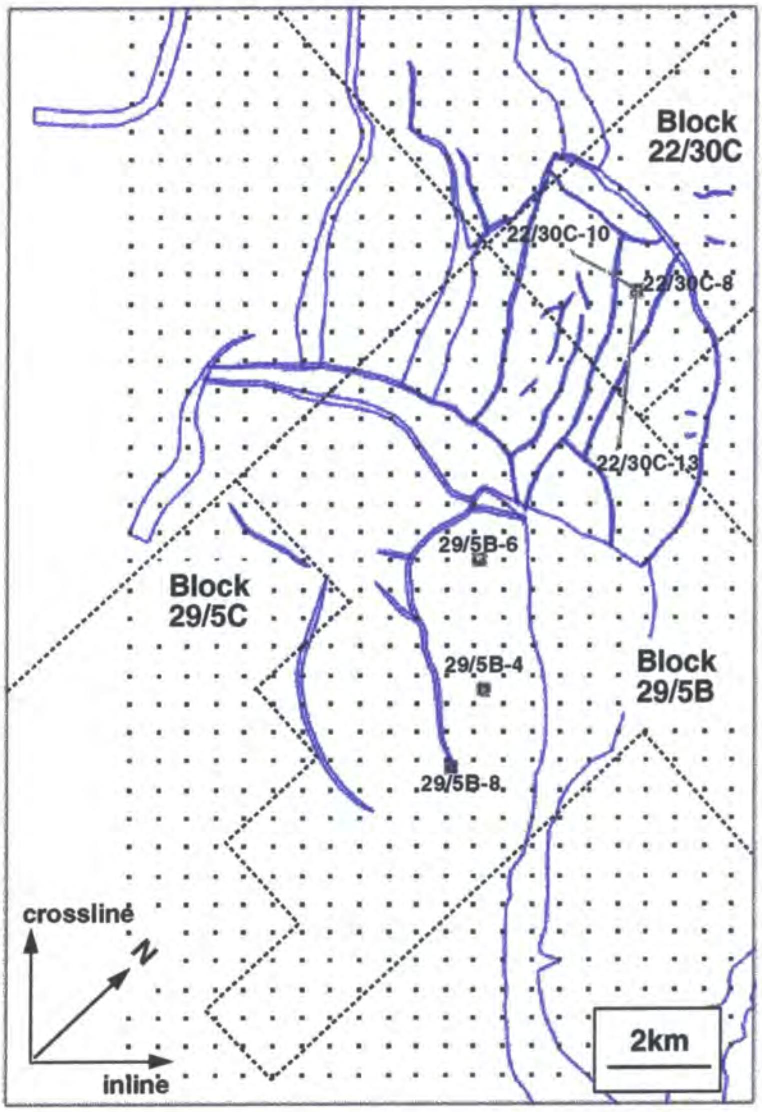


*Figure 5-6: Base map showing the region over which the inversion was carried out and the positions of the wells and the fault polygons picked for the Pentland layer*

Six-wells were used during the inversion. Well 22/30c-8 is approximately vertical and wells 22/30c-10 and 22/30c-13 are deviated wells drilled from the same surface location. Wells 29/5b-4, 29/5b-6 and 29/5b-8 are approximately vertical. All the wells have markers defined for each layer with the exception of well 22/5b-4 which is not deep enough to penetrate the Pentland. Unfortunately, no well seismic data were available.

Initially, an isotropic velocity model was built using the stacking velocities but not the well markers. Each layer required three to five iterations to converge starting from a simple constant velocity initial guess. The depth-migration stage of the inversion is relatively quick. In this case, one ray was fired for each node on the time-migrated grids which meant that around 130 000 rays were used to depth migrate each layer. This takes only a few seconds per iteration for the shallow layers and increases to around half a minute for the deepest layer on a one processor workstation[1]. The forward modelling of the stacking velocity field is the most expensive part of the inversion. The deepest layer takes around 5 minutes per iteration to forward model 2532 ray fans. The last layer, therefore, took a total time of around 30 minutes for five iterations.

As the calculation of the stacking velocity at each VA location is independent, the forward modelling is well-suited to parallelisation. Tests on a modern shared memory machine, that allows the entire velocity model to be stored in memory accessible to all processors, have shown that the speed increase as a function of the number of processors is almost linear. This is important when assessing uncertainties in the inverted result as it makes it feasible to carry out a number of inversions with different types of constraint and model parameterisation. On a 32 processor machine, therefore, the inversion of the Pentland layer takes less than a minute.

An anisotropic model was also produced by including the well marker depths into the inversion in addition to the stacking velocity field. The anisotropic ray-tracing is approximately five times slower than the isotropic due to the added complexity in the traveltime slowness calculation. Correspondingly, the anisotropic inversion was five times slower than the isotropic.

---

[1] All performance figures are based on a Silicon Graphics, Indigo workstation, with a MIPS R4400 CPU running at 250 MHz

## 5.3 Isotropic Model

The isotropic inversion results of each of the six layers, beginning with the shallowest, are shown in figure 5-7 through to figure 5-12. These plots show comparisons of the modelled and actual stacking velocity, as well as three other properties calculated during the modelling: the modelled zero-offset time, calculated by Gaussian weighted regression from the traveltime-offset curve, the time-pick error, which is the difference between the modelled time and the nearest stacking velocity pick, and the weighting, which gives a measure of the confidence in the pick (see section 3.4). There is no interpolation applied to these properties: instead, the surface is partitioned into regions around each of the zero-offset reflection points, and then these regions, the size of which depends upon the local reflection point density, are filled with a colour corresponding to the property value. This is purely for ease of visualisation and prevents the property variation from 'looking' smooth. All the properties, therefore, are displayed in their depth-migrated position. The resultant depth surface and vertical interval velocity are also shown.

The Balder layer is fairly flat and the variation in the actual stacking velocity is small (only around 120 m/s over the entire area). The variations in the actual stacking velocity are very high frequency although there is a clear low frequency trend from south to north. The modelled stacking velocity field after inversion captures this trend well, and the residual misfit is high frequency and low amplitude (the weighted RMS average misfit in the stacking velocities is only 20.8 m/s). Due to the lack of structure, the modelled stacking velocity field is very similar to the inverted velocity function. The top-Balder interface was a strong reflector and was picked horizon-consistently during the stacking velocity analysis. The time-pick error is correspondingly small and the weighting for the vast majority of VA locations high. Third-order velocity variations were considered sufficient as higher order terms failed to significantly reduce the value of the objective function. Similarly, the addition of a vertical gradient did not further reduce the objective function, although this is unsurprising given the lack of structure. The resolution matrix also showed that the vertical velocity gradient was completely unresolved by the available data.

The results of the isotropic inversion are compared with the well-marker depths in figure 5-13 and show that, by Balder, the depth surface is already around 250 metres too deep (some 7.5%). During routine industrial processing, the well-tie is made by applying a vertical stretch factor to the depth surface. Figure 5-14 shows the well-marker misfits after applying the best average stretch factor (0.925). For Balder, this simple correction fits all the wells to within 20 metres.

The Balder-Maureen interval is very thin (only around 200 m on average) and consequently the vertical RMS velocity down to the Maureen is relatively insensitive to the interval velocity in the Balder-Maureen interval. The velocity contrast between the Balder and Maureen layers is very large (hence the strength of the Balder reflection), and there is a strong refraction of rays across the Balder interface. The modelled stacking velocity field again captures the low-frequency trend present in the actual stacking velocities and the misfit is high frequency. The depth mistie at the wells has increased slightly but is still compensated for by a vertical stretch factor.

The Hod layer is fairly flat and conformal to the layer above. The modelled stacking velocity field is starting to contain some higher frequency events. Given the smoothness of the overburden velocity model, these high frequency events are structural in origin and correlate strongly with the dip of the depth surface. This strong structural dependence of the stacking velocity field shows that already, at this early stage, the DMO and pre-stack time-migration schemes have failed in their objective. Moreover, the presence of these high frequency events in the actual stacking velocities shows that the model is quantitatively predicting their failure. The weighted RMS error in the stacking velocities has increased to 41 m/s but remains predominately high frequency.

In the Hidra layer, the time-pick error is consistently negative over the whole layer, suggesting that the stacking velocity analysis was not horizon-consistent and consequently the weighting is low (in fact, reflections from a conformal layer slightly deeper than the Hidra were picked). The majority of the modelled times were within 50 ms of the nearest pick however, and interpolation of the stacking velocity field was not deemed necessary. The modelled stacking velocity field is now of considerable complexity and is very clearly related to the depth structure. There is also a very strong correlation between a low velocity zone on the inverted vertical velocity function and a shallow area of the depth structure. Such a feature could also be explained by the introduction of a vertical velocity gradient; however, a vertical gradient in addition to the lateral variations failed to further reduce the value of the objective function. The well misfits have not increased very much considering the thickness of the interval, suggesting the velocity field to be largely isotropic.

The structure of the BCU is considerably more complex than the layers above and was heavily faulted. The fault planes themselves were not imaged well on the seismic data and were not interpreted reliably. In the first stage of the inversion, therefore, points that were inside the fault polygons were removed and consequently the depth-migrated surface contained many holes. These gaps must be filled for the depth-migration of deeper layers and to produce a grid of interval velocities for subsequent depth-migration of the seismic data. I have devised an automatic procedure that makes use of the connectivity relationships between the points in the depth domain and fills any gaps with the smoothest possible surface. As points from within the fault planes were removed prior to demigration, zero-offset ray solutions reflecting off the fault-planes were not predicted and not traced during subsequent inversion. This speeds up and stabilises the inversion. For the deeper horizons the RMS velocity is significantly higher than the 2000 m/s used for the pre-stack time migration. When combined with the large time-dips present on these layers, this means that the velocity analysis was carried out on strongly under-migrated data with residual triplications, and consequently many of the VA locations still have several solutions. Without pre-stack time migration, however, there would be significantly more zero-offset solutions (around four times as many, in this case) and the forward modelling would be considerably slower.

The Pentland layer was also heavily faulted and reflections from the faults removed using the same procedure as described above. Due to the complexity of the Pentland layer and the increase in noise, the stacking velocity field could not be picked horizon-consistently. Instead, stacking velocity picks were made along constant time slices. During inversion, therefore, the stacking velocity function had to be interpolated and there was no time-pick error. The modelled and actual velocity fields are very similar, and although the RMS error in the stacking velocities increased to 98.8 m/s, this was not considered excessive given the noise and the wide range of stacking velocities over the layer. Importantly, an average vertical stretch factor is no longer sufficient to tie all the well markers to the required accuracy (the well 29/5b-8 has an error of nearly 97 m after correction). Of course, the well tie could be made by producing a laterally varying map of stretch factors, but such a process is highly subjective and strongly non-unique. Despite the relief, a vertical gradient could not be resolved from the available data in conjunction with lateral variations.

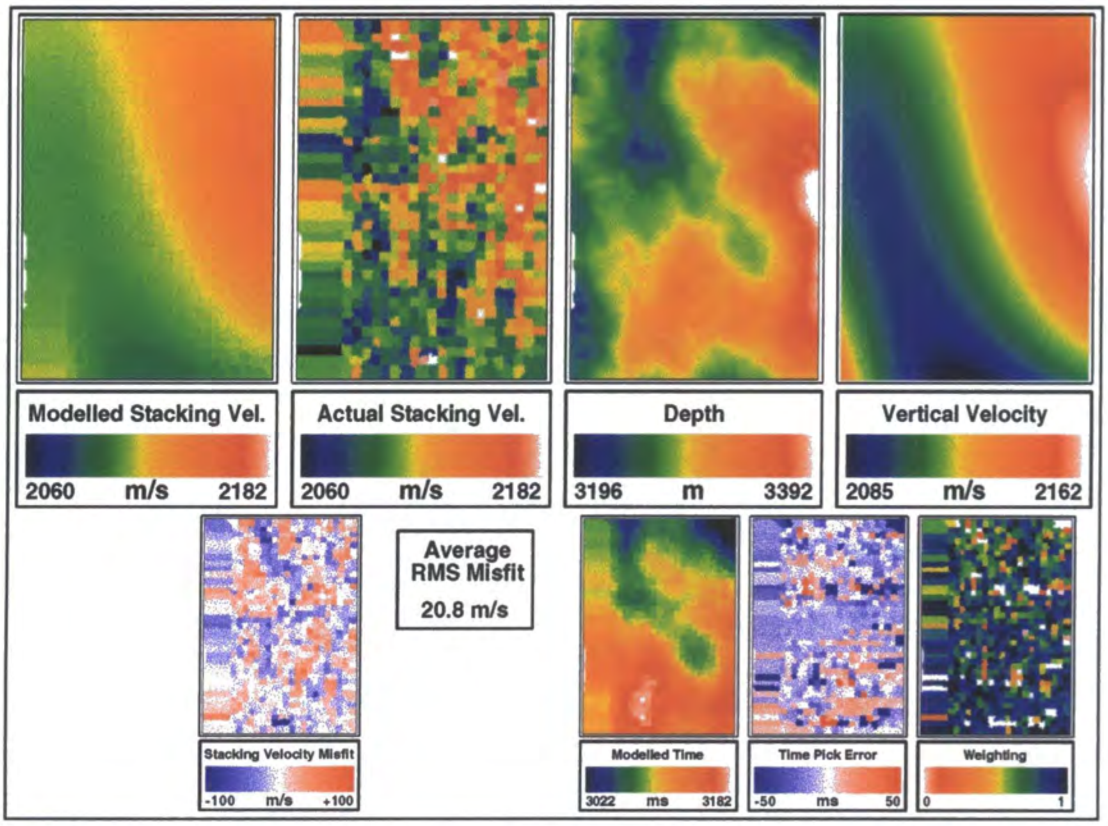


*Figure 5-7: Results from the isotropic modelling of Balder*

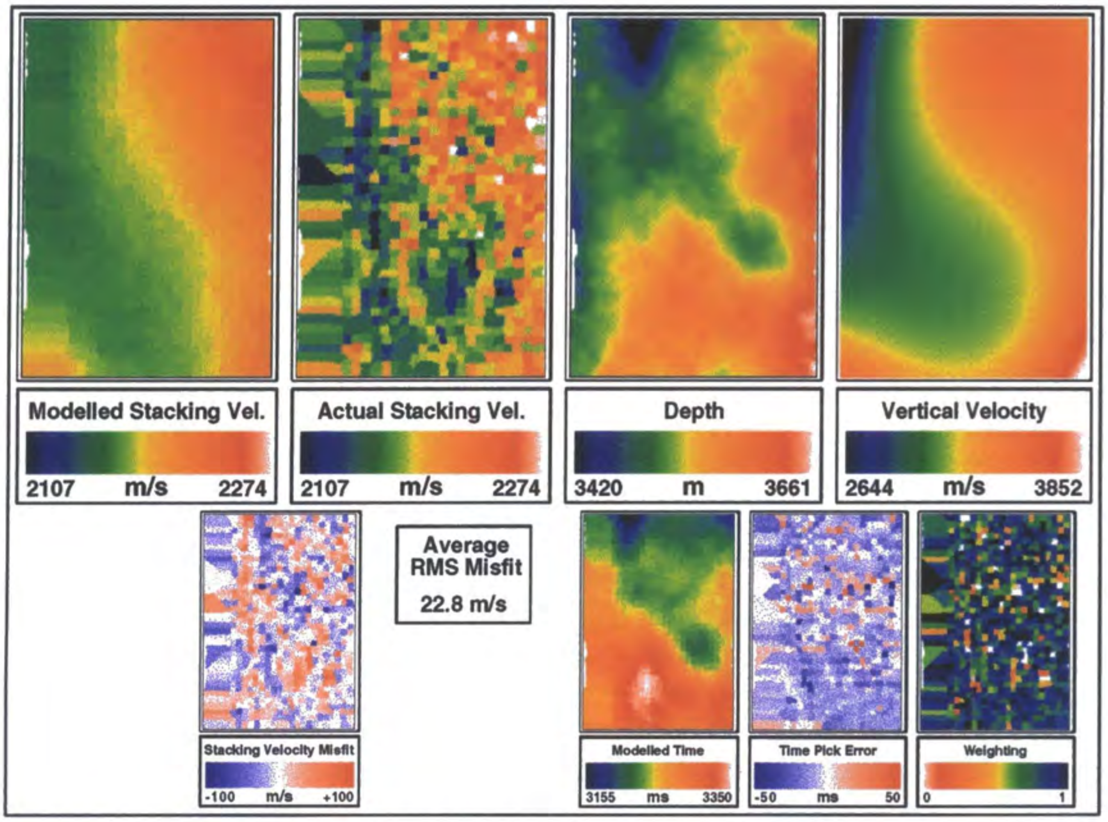


*Figure 5-8: Results from the isotropic modelling of Maureen*

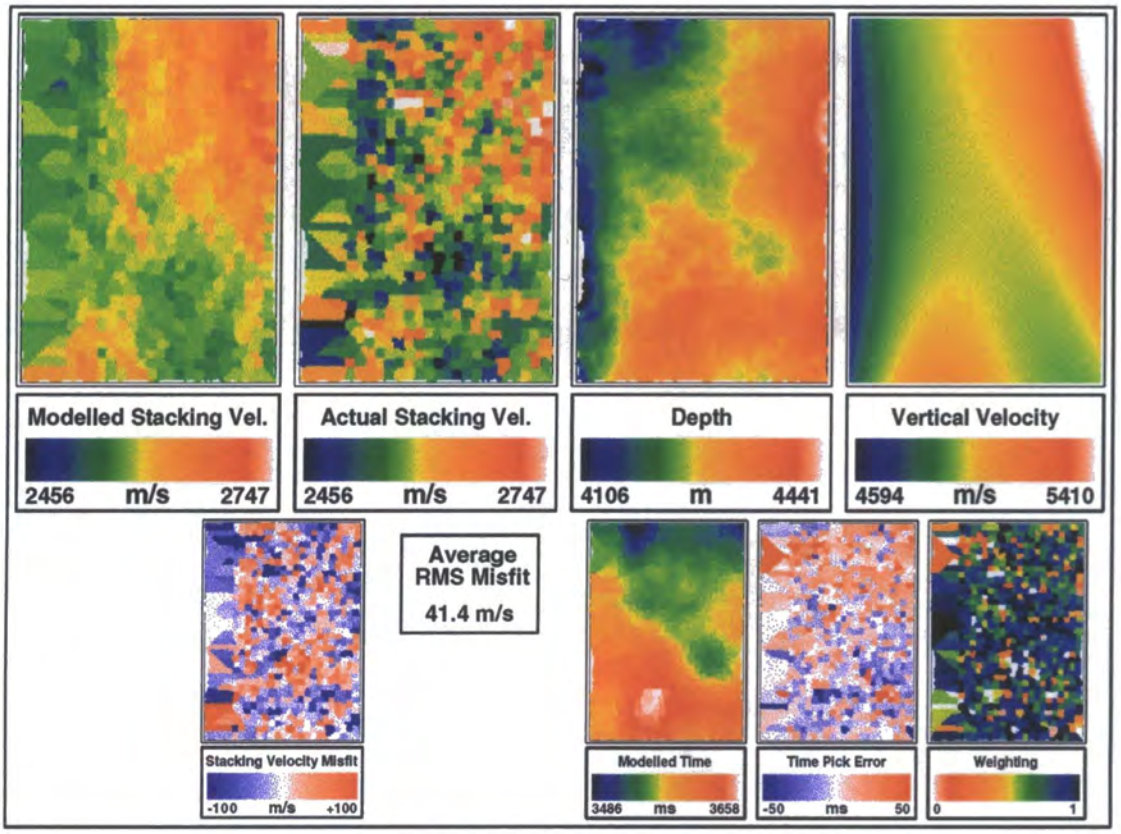


*Figure 5-9: Results from the isotropic modelling of HOD*

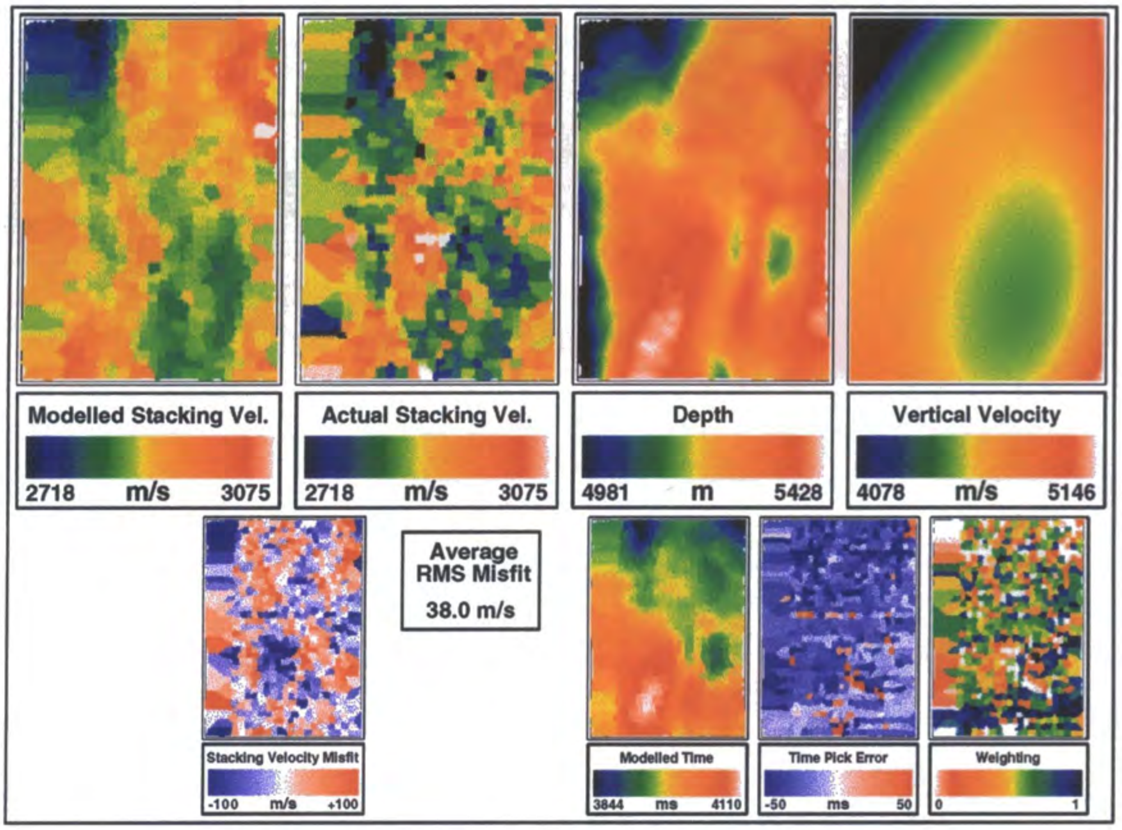


*Figure 5-10: Results from the isotropic modelling of Hidra*

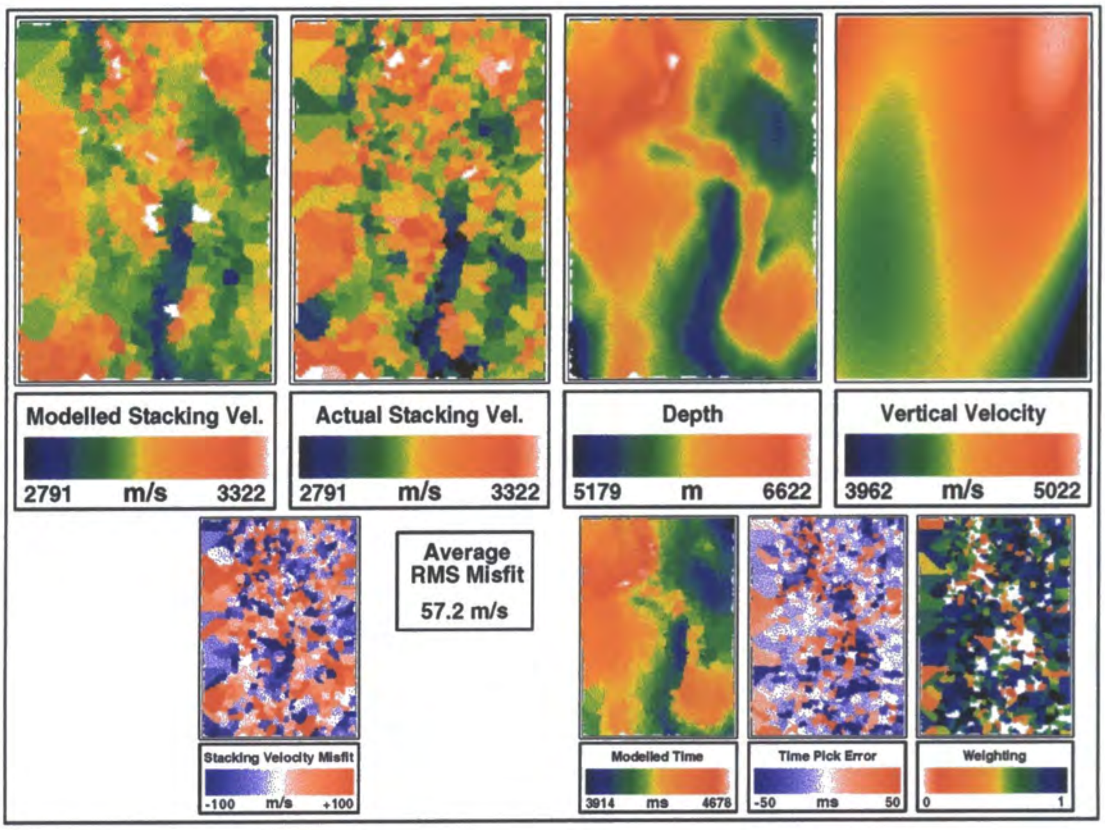


*Figure 5-11: Results from the isotropic modelling of BCU*

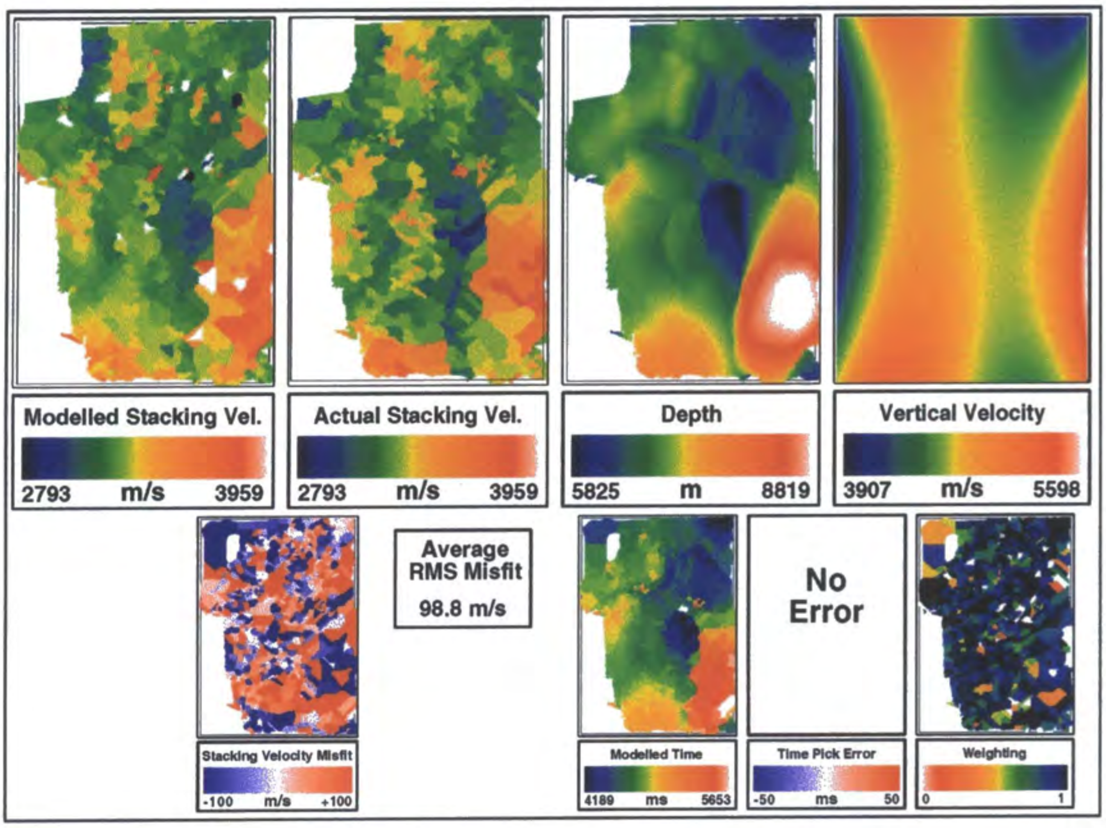
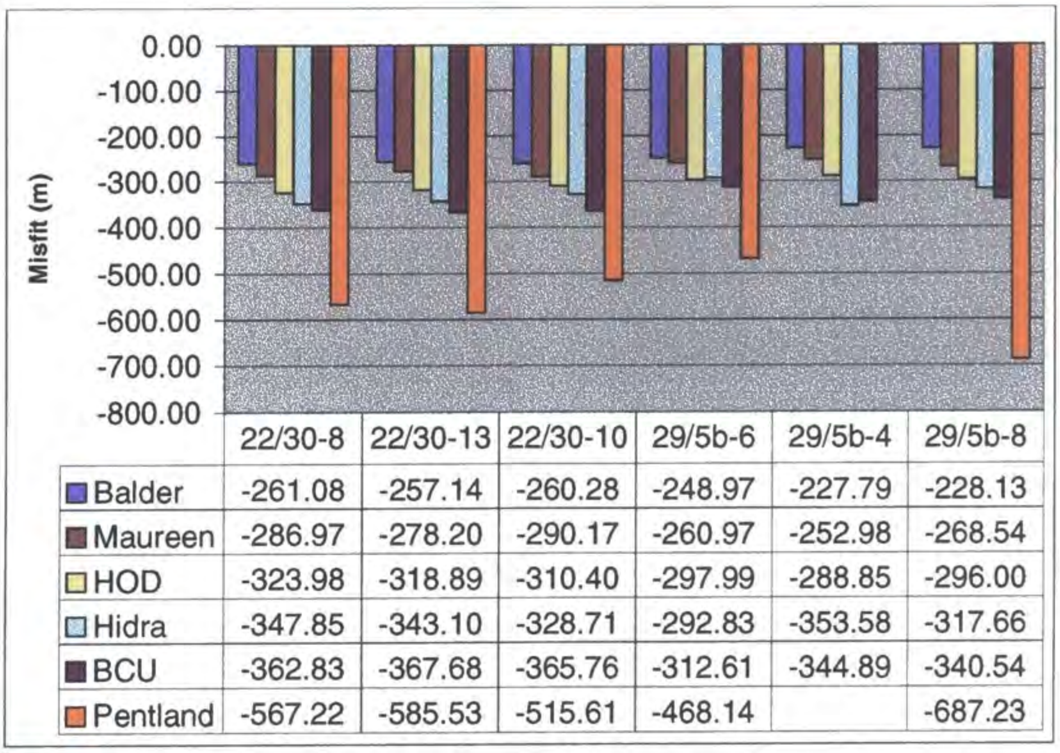
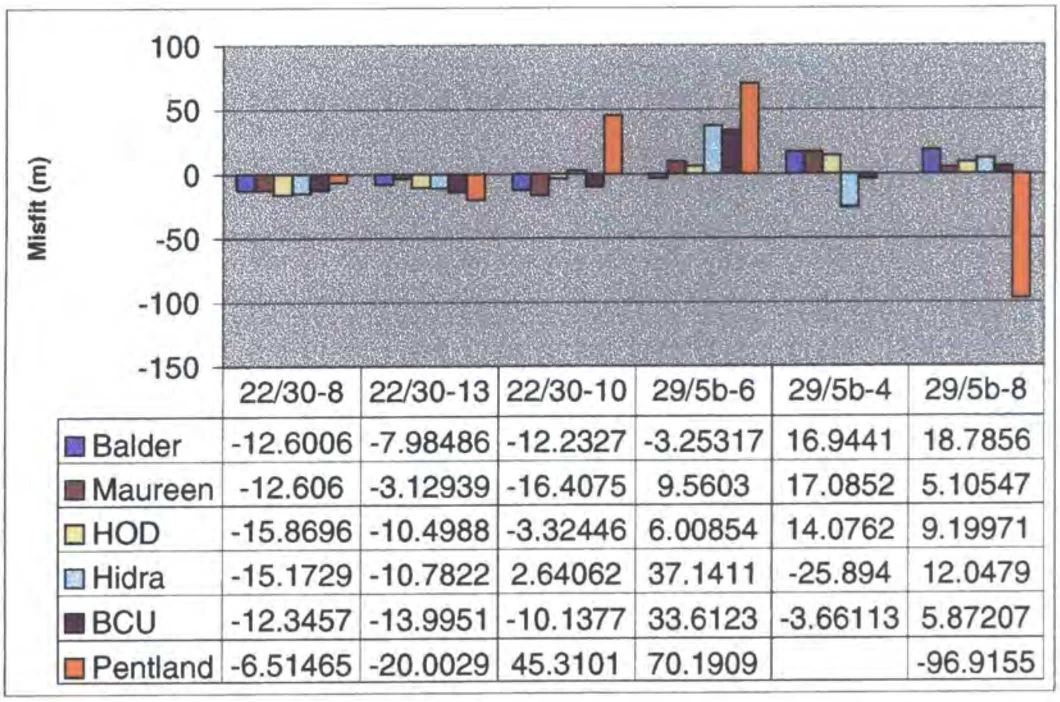


*Figure 5-12: Results from the isotropic modelling of Pentland*



| | 22/30-8 | 22/30-13 | 22/30-10 | 29/5b-6 | 29/5b-4 | 29/5b-8 |
|---|---|---|---|---|---|---|
| 🟦 Balder | -261.08 | -257.14 | -260.28 | -248.97 | -227.79 | -228.13 |
| 🟫 Maureen | -286.97 | -278.20 | -290.17 | -260.97 | -252.98 | -268.54 |
| 🟨 HOD | -323.98 | -318.89 | -310.40 | -297.99 | -288.85 | -296.00 |
| 🟦 Hidra | -347.85 | -343.10 | -328.71 | -292.83 | -353.58 | -317.66 |
| 🟪 BCU | -362.83 | -367.68 | -365.76 | -312.61 | -344.89 | -340.54 |
| 🟧 Pentland | -567.22 | -585.53 | -515.61 | -468.14 | | -687.23 |

*Figure 5-13: Isotropic well misfits at each layer*

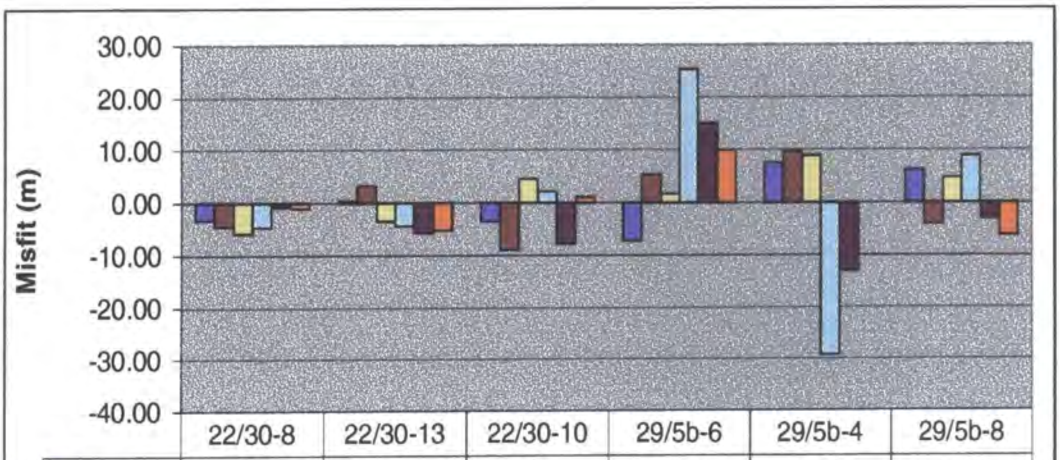


| | 22/30-8 | 22/30-13 | 22/30-10 | 29/5b-6 | 29/5b-4 | 29/5b-8 |
|---|---|---|---|---|---|---|
| 🟦 Balder | -12.6006 | -7.98486 | -12.2327 | -3.25317 | 16.9441 | 18.7856 |
| 🟫 Maureen | -12.606 | -3.12939 | -16.4075 | 9.5603 | 17.0852 | 5.10547 |
| 🟨 HOD | -15.8696 | -10.4988 | -3.32446 | 6.00854 | 14.0762 | 9.19971 |
| 🟦 Hidra | -15.1729 | -10.7822 | 2.64062 | 37.1411 | -25.894 | 12.0479 |
| 🟪 BCU | -12.3457 | -13.9951 | -10.1377 | 33.6123 | -3.66113 | 5.87207 |
| 🟧 Pentland | -6.51465 | -20.0029 | 45.3101 | 70.1909 | | -96.9155 |

*Figure 5-14: Isotropic well misfits after correction with the average vertical stretch factor*

## 5.4 Anisotropic Model

An anisotropic model was also built using the stacking velocities and the well-marker depths. The results of the inversion for each layer are presented in figure 5-16 through to figure 5-21 and the well marker misfits are shown in figure 5-15. For ease of comparison the colour scales for the stacking velocity field have been chosen to be the same as for the isotropic results. The introduction of anisotropy makes it possible to produce a velocity model that is consistent with the stacking velocities and the well marker depths. The majority of the well markers are honoured to within 10 m (figure 5-15). This is considerably better than the result obtained by vertical stretching of the isotropic model, especially for the Pentland layer, where an average vertical stretch was insufficient. The improvement over the vertical stretch is possible because the wells intersect the depth surface at a range of dips and, therefore, a constant set of anisotropy parameters results in different propagation velocities for the depth migration rays around the different wells.



| | 22/30-8 | 22/30-13 | 22/30-10 | 29/5b-6 | 29/5b-4 | 29/5b-8 |
|---|---|---|---|---|---|---|
| ■ Balder | -3.29 | 0.31 | -3.51 | -7.19 | 7.41 | 6.18 |
| ■ Maureen | -4.54 | 3.28 | -8.93 | 5.31 | 9.59 | -4.08 |
| □ HOD | -5.83 | -3.45 | 4.49 | 1.68 | 8.83 | 4.70 |
| □ Hidra | -4.64 | -4.42 | 2.19 | 25.33 | -29.29 | 8.78 |
| ■ BCU | -0.73 | -5.76 | -7.81 | 15.01 | -13.00 | -2.99 |
| ■ Pentland | -1.07 | -5.29 | 1.07 | 9.89 | | -6.14 |

*Figure 5-15: Anisotropic well misfit at each layer*

The addition of anisotropy does not improve the fitting of the stacking velocity field as measured by the weighted RMS stacking velocity misfit. This suggests that, in this case, there is very little information about the anisotropy from the stacking velocities alone. Given that the zero-offset traveltime to each layer is not dependent upon the velocity field, the modelled traveltime is almost the same for the isotropic and anisotropic case (there is some small difference due to the non-hyperbolicity of the traveltime curve), and consequently the actual stacking velocity field is very similar to that in the isotropic case.

The inverted anisotropy parameters within each layer are fairly large with the exception of the Hod-Hidra interval which has $\delta = 0.019$ and $\varepsilon = 0.026$. Interestingly, this interval is a fairly homogeneous chalk layer which the gamma ray well logs shows to have a low shale content. Conversely, the other layers with large inverted anisotropy values contain thick shale beds. This correlation between the level of anisotropy and the lithology of the layers is in agreement with the work of Banik (1984) who demonstrated that the well misfit after isotropic Dix inversion was strongly correlated with the presence of thick shale intervals. Although the stacking velocities in addition to the well markers indicate the presence of anisotropy, the resolution matrix shows that both anisotropy parameters are not resolved in any of the layers. For this model, a constraint of $\varepsilon = 2\delta$ (thought to be representative of shale) was imposed to remove the non-uniqueness and select the required solution. The uncertainty associated with this non-uniqueness can be assessed by making several models with different constraints.

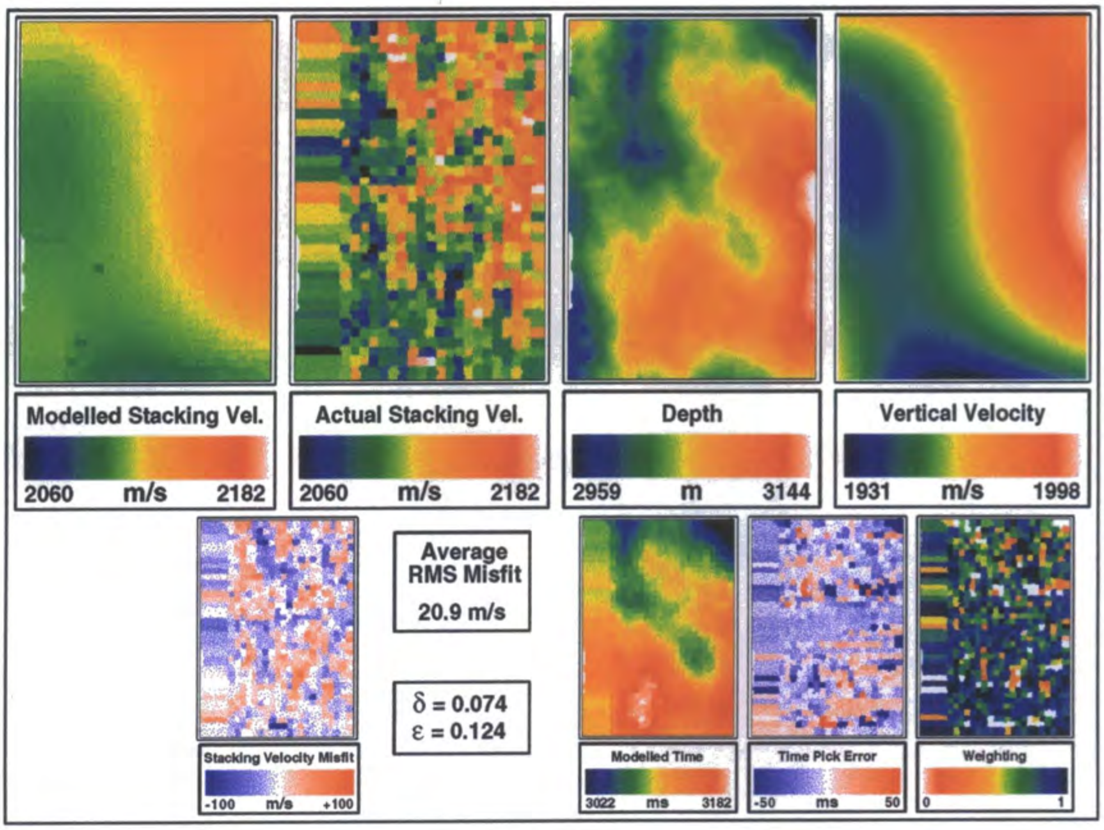


*Figure 5-16: Results from the anisotropic modelling of Balder*

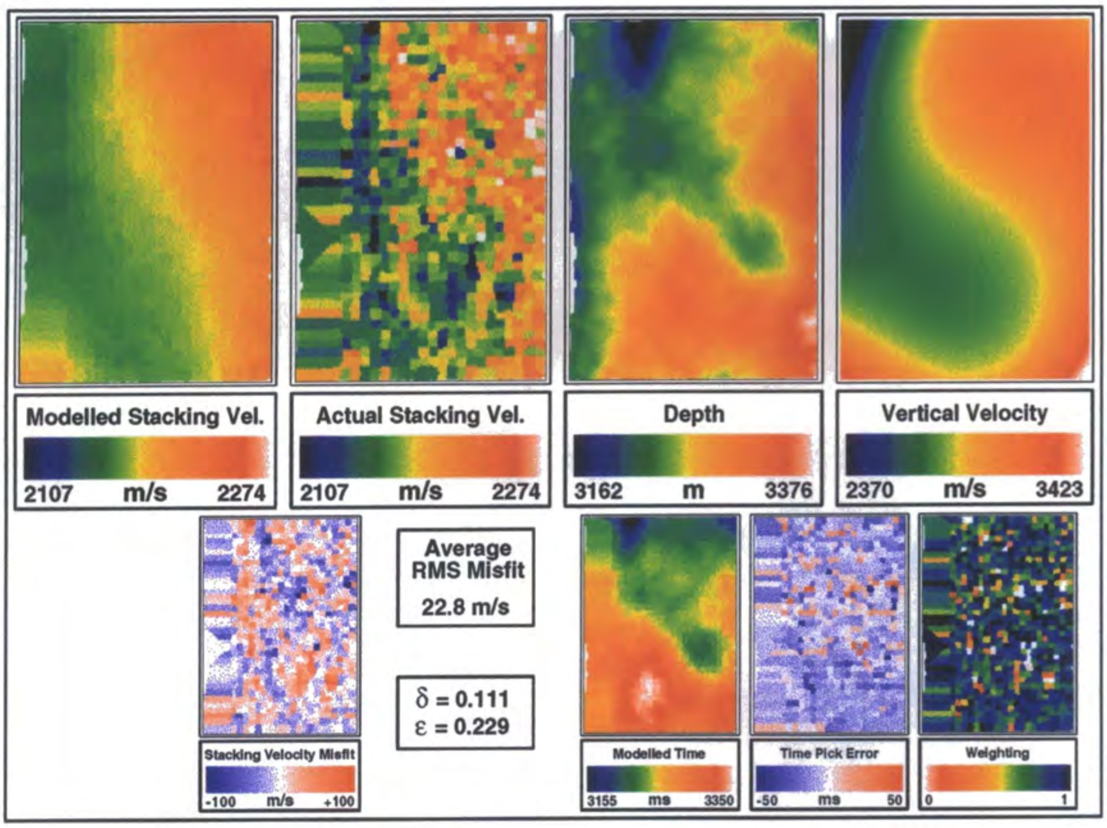


*Figure 5-17: Results from the anisotropic modelling of Maureen*

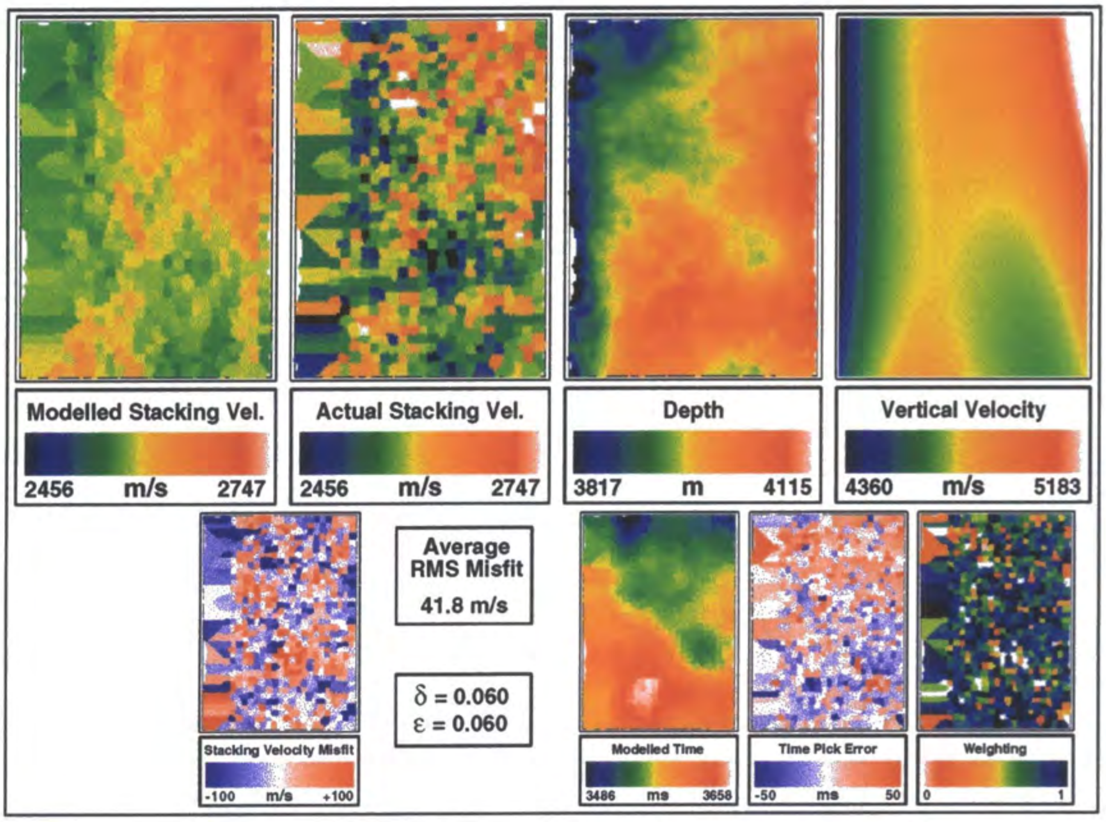


*Figure 5-18: Results from the anisotropic modelling of HOD*

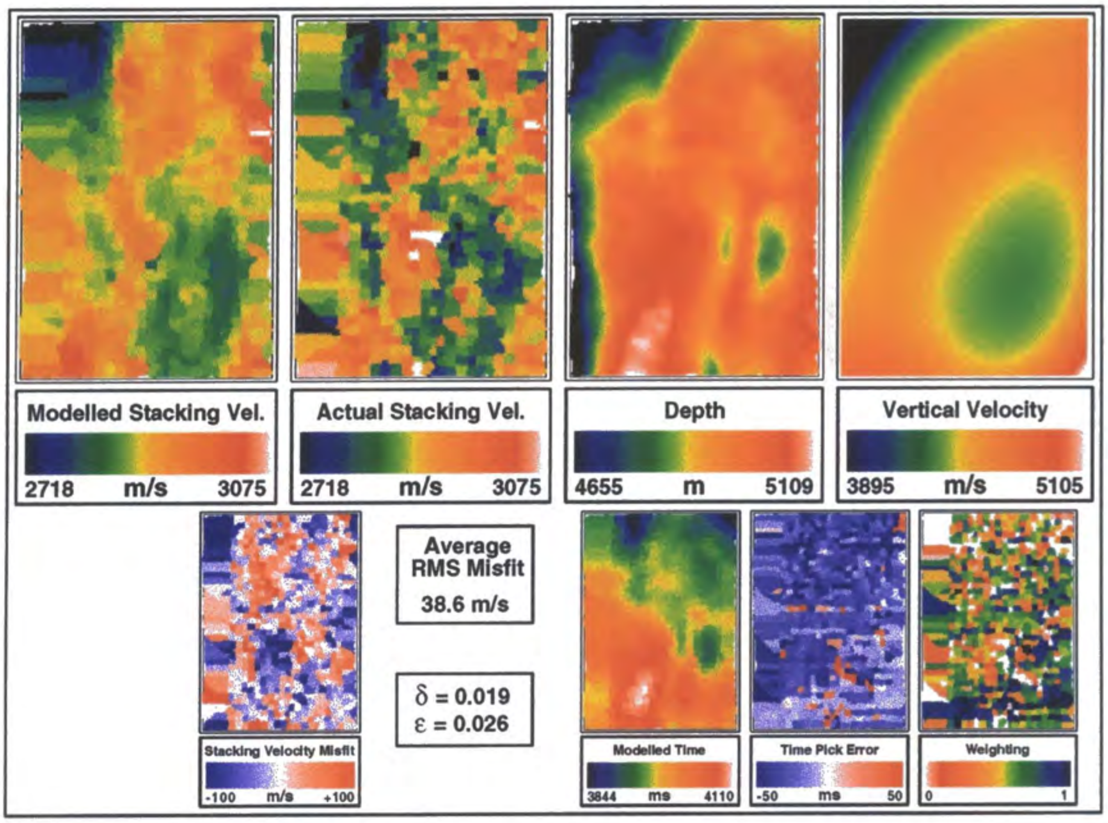


*Figure 5-19: Results from the anisotropic modelling of Hidra*

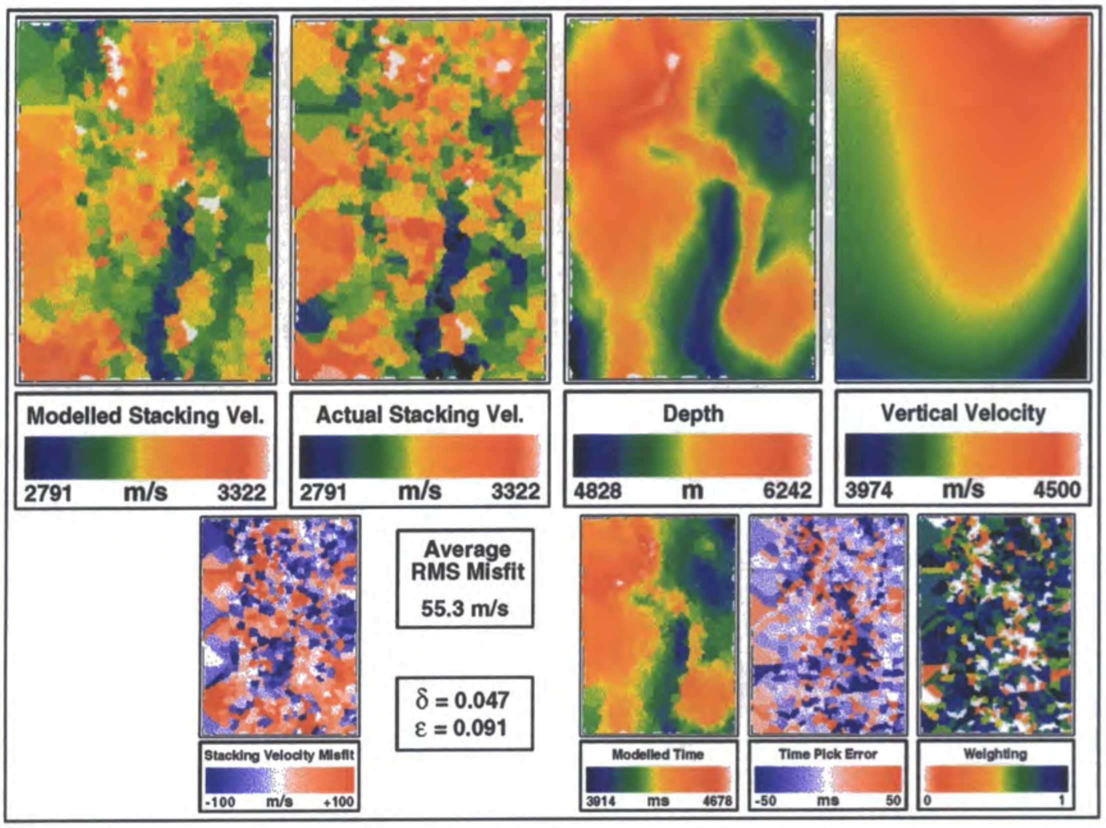


*Figure 5-20: Results from the anisotropic modelling of BCU*

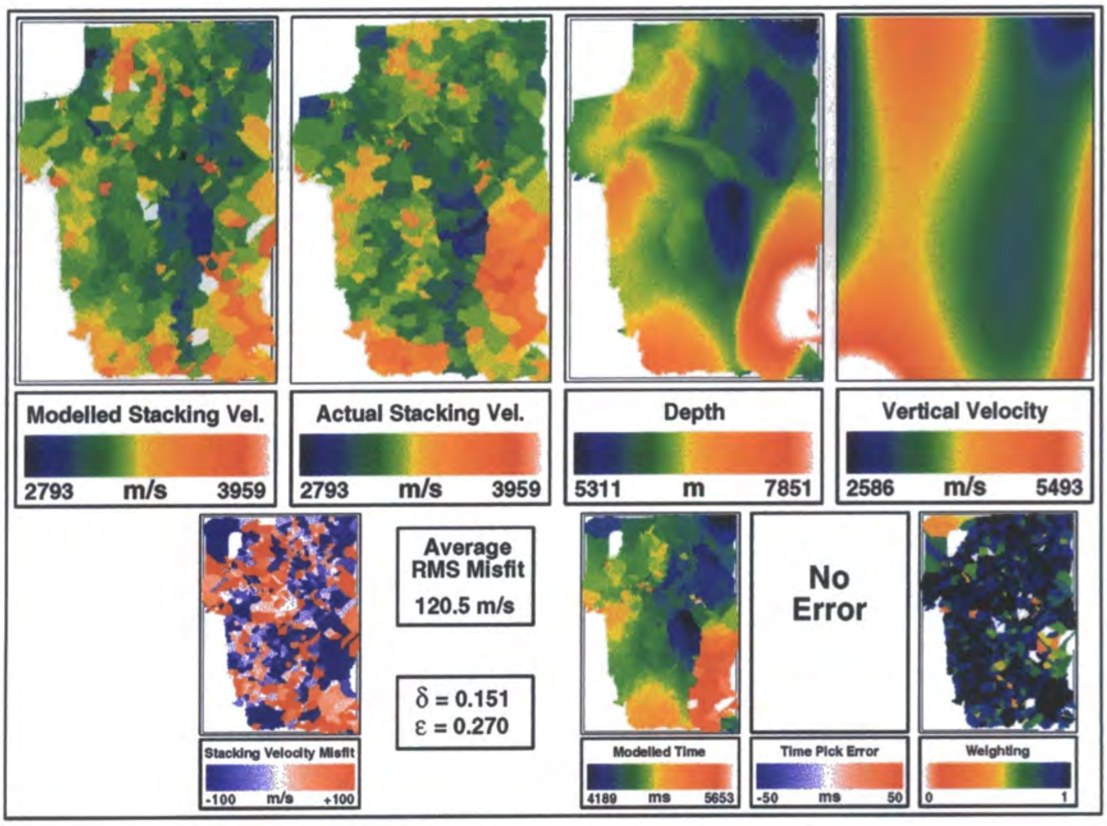


*Figure 5-21: Results from the anisotropic modelling of Pentland*

For the shallower layers, the shape of the inverted vertical velocity function within each layer is very similar in shape to the isotropic result although the values of the vertical velocities are all much smaller. The resolution matrix shows the isotropic and anisotropic parameters are resolved almost independently. For the deepest three layers, despite the structure, the anisotropy parameters remain poorly resolved and contain trade-offs with the higher-order lateral variation velocity terms. This results in the changes of shape of the vertical velocity functions between the isotropic and anisotropic models in the deeper layers.

Figure 5-22 shows a 3D view of the inverted anisotropic model in depth. The depth surfaces for all six inverted horizons are shown. The Balder, Maureen and Hod layers are all fairly flat, the Hidra has some gentle structure and the BCU and Pentland layers are heavily faulted. The Pentland layer is very deep with a depth range of 5311 to 7851 m below mean sea level. The three Elgin wells drilled from the same surface location and the three, almost vertical, Franklin wells are also shown. The wells all intercept the Pentland layer at its shallowest points. This highlights the difficulty of using information from the well logs in order the estimate interval parameters over the entirety of the layer. The estimations of the vertical velocity gradient from the well logs, for example, could not be expected to be representative of the velocity variation between the BCU and Pentland in the areas away from the wells, where the interval thickness is much greater and the top Pentland is much deeper.

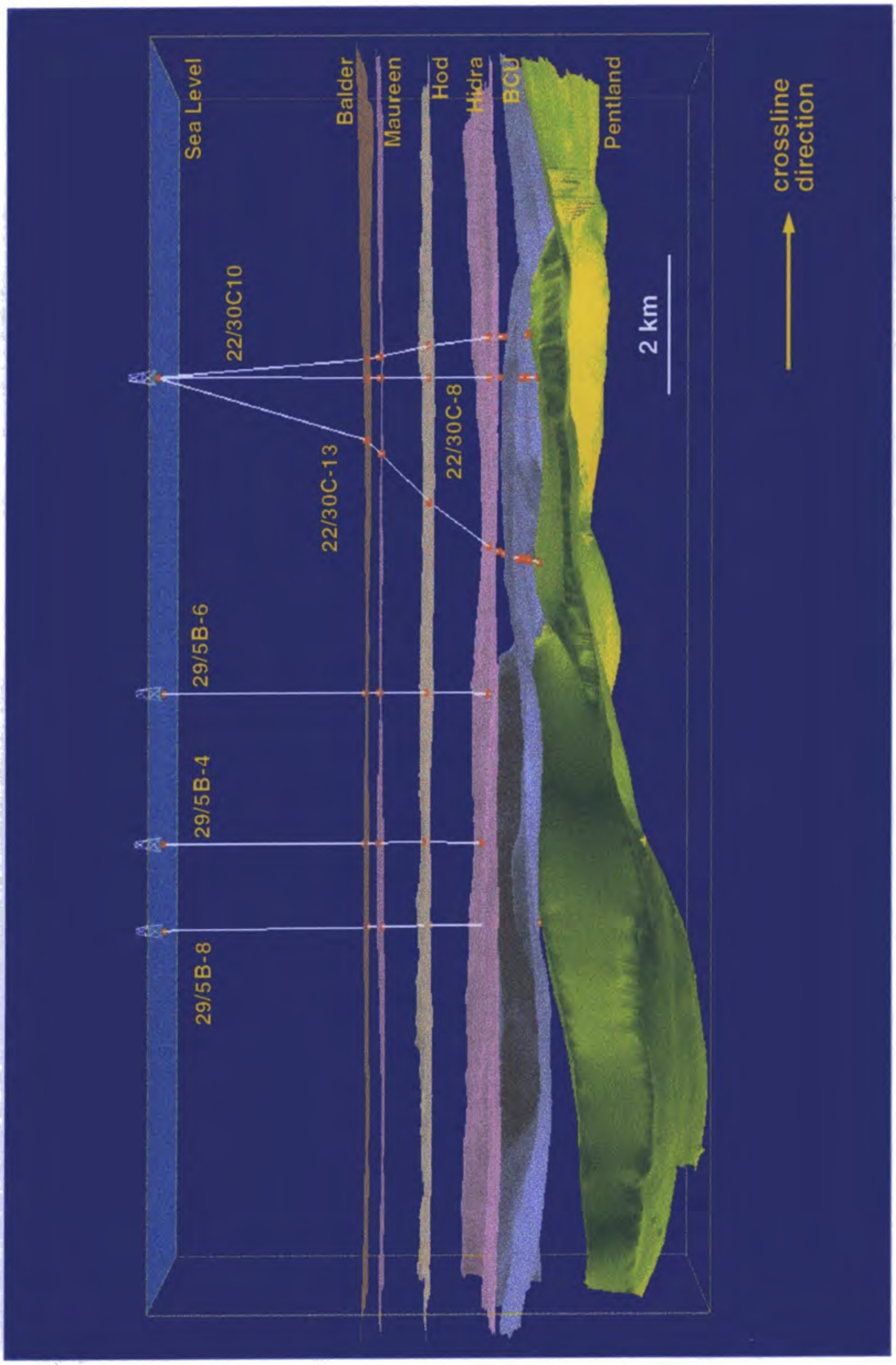


*Figure 5-22: A 3D view of the anisotropic depth model showing all layers and wells*

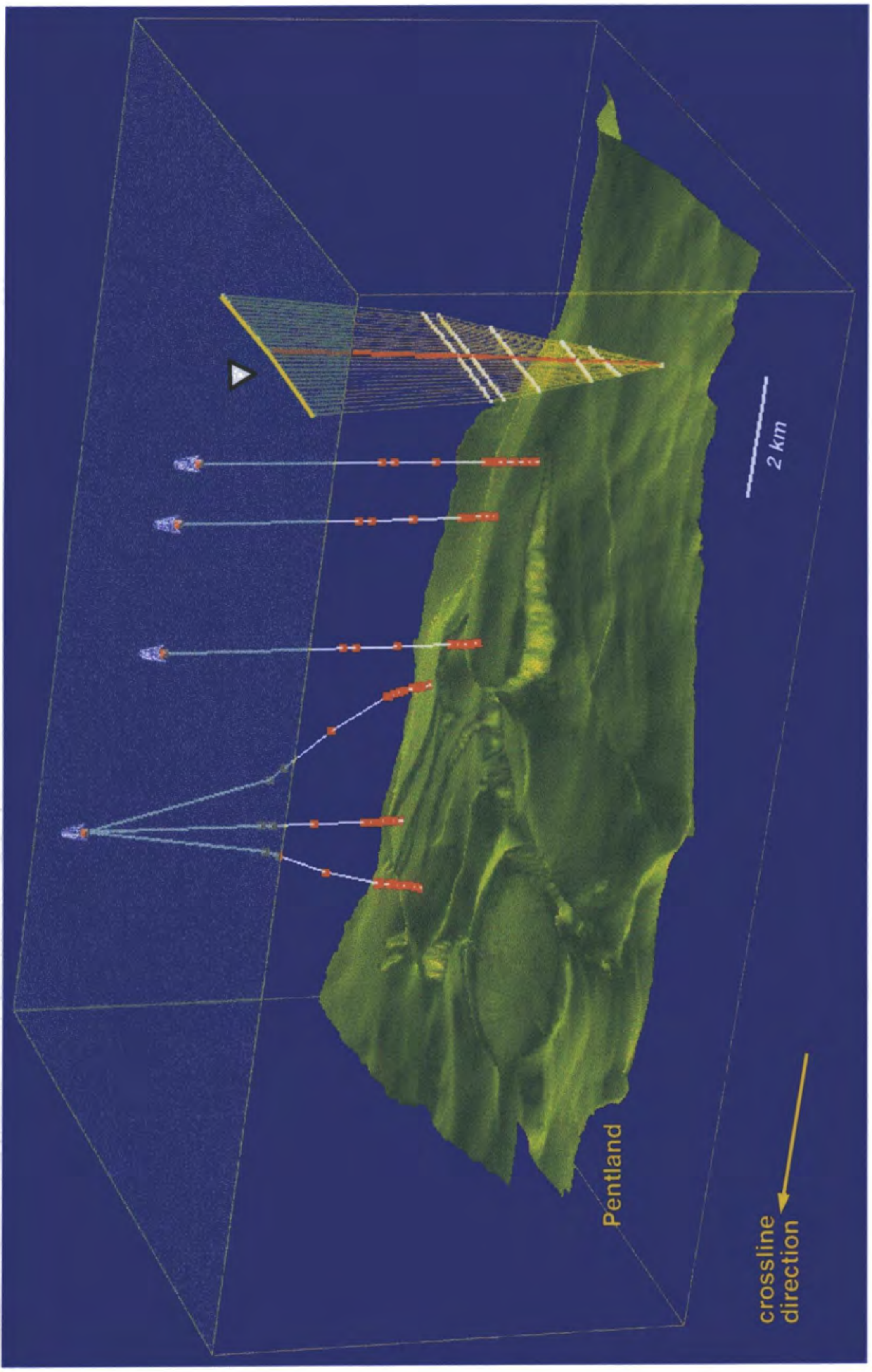


*Figure 5-23: A 3D view of the depth map for the Pentland layer showing the quality of the imaging*

Figure 5-23 shows a 3D view of the Pentland depth surface and demonstrates the quality of the depth maps produced by the inversion. The faults are very sharply defined. There is an interesting effect to the north-east of the three Franklin wells, bordering the large fault (this fault is shown more clearly in figure 5-22). The fault surface intersects the BCU layer, as well as the Pentland, and the zero-offset rays used in the map migration must cross this fault. The effect of this is visible in figure 5-23 as a zone of discontinuous behaviour referred to as a fault shadow. Imaging in these areas is very difficult using ray theory as the velocity model is probably not representative of the true velocity variation over the fault. A single stacking velocity offset fan is also shown. A strong refraction, due to the large velocity contrast at the Balder interface, is visible. Also, as the velocity analysis was after PreSTM, the zero-offset ray does not leave the datum surface from the VA binning location.

## 5.5 Comparison of the Isotropic and Anisotropic Models

The difference between the time- and depth-migration can be assessed visually by means of a migration distance plot (figure 5-24 and figure 5-25). These figures show the lateral displacement between the time- and depth-migrated domains for both the isotropic and anisotropic models at the Pentland layer. The position of the well in the time-migrated domain (as discussed in section 2.10) is also shown. Such plots are extremely useful quality control tools as they show the areas where the time-migration operator is not working well, highlighting areas that may need reinterpretation. Operationally, they are invaluable as they show the error that is made by a simple vertical stretch of the time-migrated interpretations.

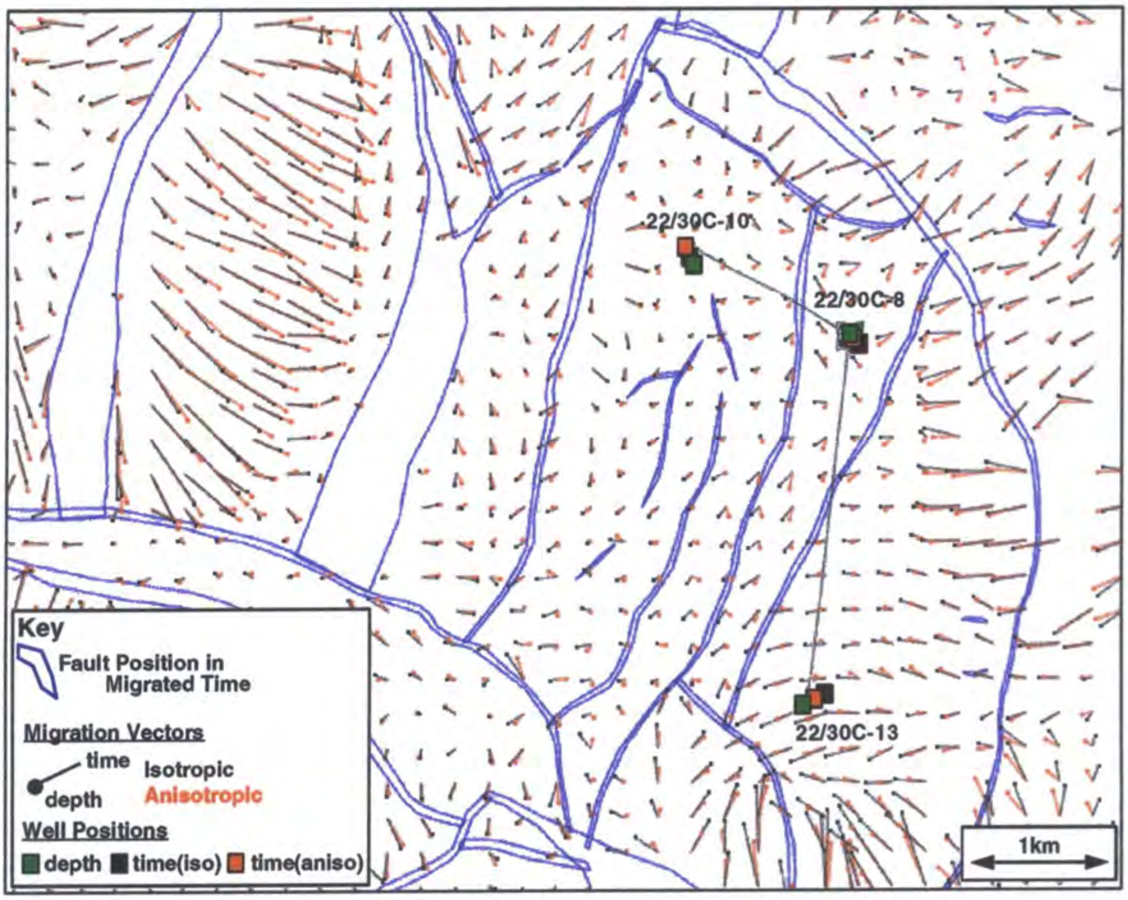


*Figure 5-24: Migration distance plot over the Elgin field showing the lateral displacements between the time-migrated and depth-migrated positions at the Pentland level*

The migration distances are not negligible as they are in excess of 500 m in the dipping areas. It is also clear that there are significant positioning differences between the isotropic and anisotropic models, both in terms of the magnitude of the lateral displacement and its direction. In this case, the time-migration velocity model was selected very carefully in order to optimise the quality of the time-migrated image. Generally, with a less sophisticated time-migration field, the migration distances could be much larger.

In the regions around the wells, where the dip is smaller, the migration distances are still not negligible and, moreover, the position of the well in the time-migrated block is different for the isotropic and anisotropic models. This shift between the position of the well in depth and the time-migrated block was shown to be important in this case.

Synthetic seismograms were calculated for each well using an estimate of the seismic wavelet and a reflectivity profile calculated from the well logs. When the synthetic seismograms were compared with the time-migrated surface seismic data, at the position of the well's trajectories in depth, large misties were observed. When the ties are made at the well's position in migrated-time, the mistie is effectively removed.

*Figure 5-25: Migration distance plot over the Franklin field showing the lateral displacements between the time-migrated and depth-migrated positions at the Pentland level*

Figure 5-26 shows the vertical difference between the anisotropic model and the isotropic model after vertical stretching to tie the wells with the average stretch factor. The differences between the isotropic and anisotropic models exceed several hundred metres in some areas. This discrepancy results in a gross rock volume difference of almost 8%.

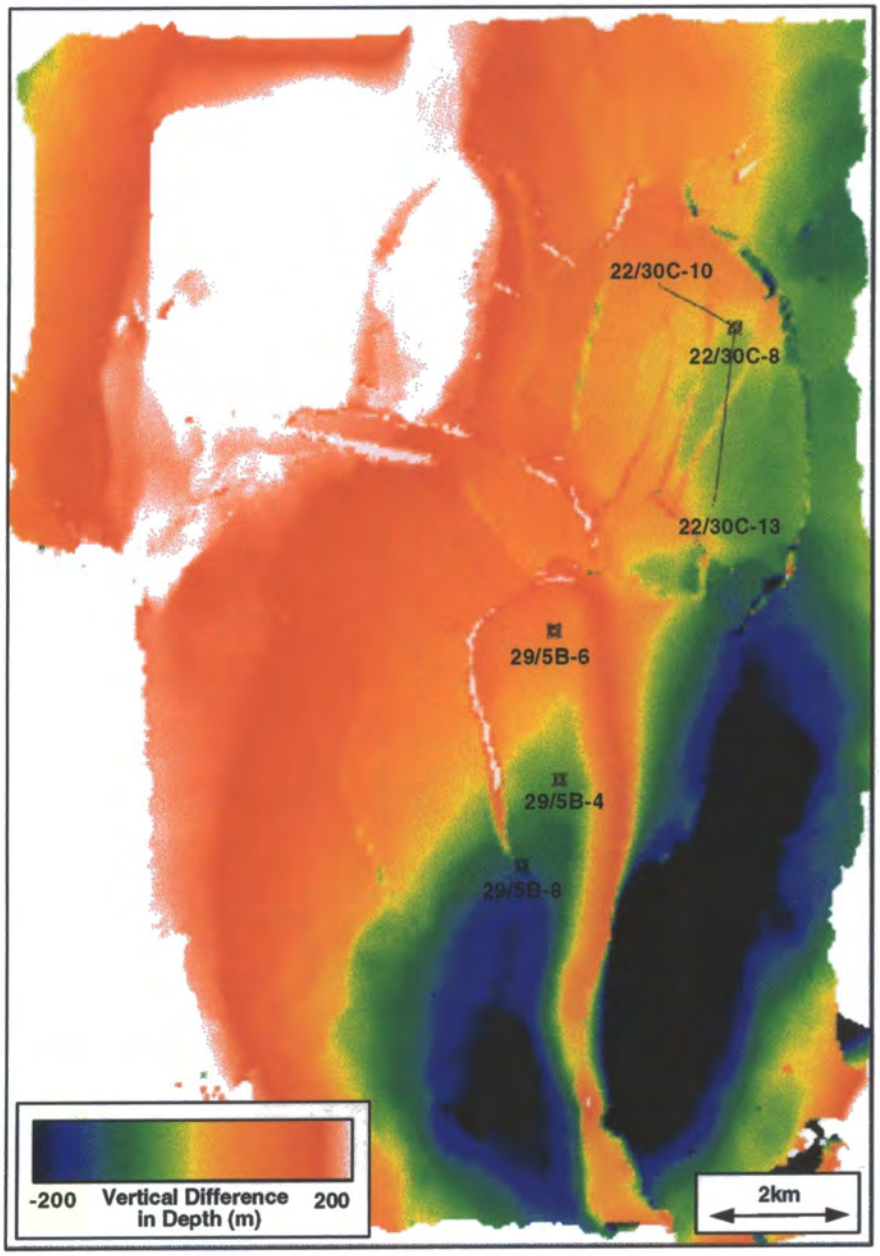


*Figure 5-26: The vertical displacement between the anisotropic depth map and the isotropic map vertically stretched to tie the wells using the average stretch factor of each well*

# 6.0 Conclusions and Suggestions for Future Work

## 6.1 Conclusions

Accurate estimation of subsurface interval velocities is essential for both structural imaging and reservoir characterisation. The use of a model-based inversion scheme has been shown to be significantly more accurate than the analytical schemes that currently dominate industrial practice. This thesis has demonstrated that such techniques can be applied quickly and robustly enough to be of real operational importance. Generally, in order to produce a velocity-depth model that ties both the surface seismic and well data the velocity model must be anisotropic. Moreover, the inclusion of anisotropy in the model is necessary for optimal seismic imaging and determination of the structure away from the wells. Unfortunately, however, the anisotropy parameters cannot usually be resolved from surface seismic reflection data alone. The addition of well seismic traveltimes is demonstrated to improve parameter resolution.

### 6.1.1 Domain Mapping

Accurate time to depth conversion requires interpreted zero-offset events. Due to the relative insensitivity of a near-offset stack to the stacking velocity field, the assumption of equivalence between the stacked and zero-offset domains is surprisingly robust and interpreted events in the stacked domain can be assumed to be zero-offset with a high degree of accuracy. Interpretation in the stacked-domain, however, is extremely difficult, especially in 3D, because it is not a good representation of the subsurface structure. Due to these problems, interpretation is routinely carried out in the time-migrated domain and so, in practice, it is these time-migrated maps that must be converted into depth.

Techniques, such as vertical stretch and image ray, that try to directly relate the time- and depth-migrated domains commonly make invalid assumptions about the time-migration procedure and are not applicable when moderate lateral velocity variations exist. Instead, the demigration of the time-migrated horizons prior to accurate ray-trace migration is an attractive way to perform depth migration. It has the speed of an interpretation based approach but does not make strong assumptions about the validity of the time-migration velocity model. The approach is limited when the complexity of the subsurface is too great or when the time-migration velocity field is too far from the optimum. The validity of the time-migration simulation should always be checked by comparing the demigrated maps with the stacked seismic section, and the inconsistencies may be used to help re-interpret the time-migrated domain when it is deemed necessary. This approach is a powerful aid to interpretation and can significantly reduce the uncertainty involved in the interpretation process. This makes interaction between the interpreter and the geophysicist essential.

Due to the different ways in which lateral variations in the migration velocity are approximated, it is not possible to define a simulation technique that is exactly valid for all time-migration algorithms. In particular, the Kirchhoff and finite-difference algorithms are shown to give significantly different results when there are strong lateral variations. The Kirchhoff operator is simulated analytically and the finite-difference operator by anisotropic ray-tracing in the time-migrated domain. The simulations have been calibrated against the real processing responses and the kinematics agree perfectly. Generally, unless strong lateral velocity variations exist in the time-migration velocity model, the Kirchhoff algorithm is used because it is significantly cheaper to apply.

The domain mapping algorithms all depend strongly upon dip. The dips are most easily calculated using a central differencing scheme. This approach breaks down, however, when the curvature of the surfaces becomes too great. As a rule of thumb, the surfaces should be smoothed so as to ensure that the highest spatial frequency is less than half the Nyquist frequency of the grid. This means that high frequency events such as faults must be explicitly defined during the interpretation stage.

An important application of the domain mapping algorithms is the transformation of the well trajectory into the migrated-time block. To perform this mapping, a two-step process is proposed. Firstly, normal rays are used to transform the well trajectory from depth into the zero-offset, or stacked domain, and then the time-migration process is simulated using either Kirchhoff or finite-difference simulation. It is clear that such a mapping requires a velocity / depth model that ties the well. This domain mapping is shown to be necessary when calibrating the surface seismic to the synthetic seismogram calculated at each well. If the comparison is made at the position of the well trajectory in depth, a significant mistie can be found.

### 6.1.2 Forward Modelling

The accuracy of a given depth-migration velocity model can be assessed by kinematically modelling the seismic acquisition and processing and comparing the results of this modelling with the actual data. The modelled data consist of a stacking velocity field, well seismic traveltimes, and the depths of the interfaces at the positions of the wells. The stacking velocity corresponding to a given reflector is found by tracing a fan of rays spanning the true range of offsets present in the data. A Gaussian-weighted regression scheme is then used to calculate the stacking velocity and stack time, given the modelled traveltime-offset curve. This is close to simulating the way in which semblance analysis actually works and is fairly insensitive to outliers which strongly effect both linear and hyperbolic regression techniques. The Gaussian-weighted regression also gives a measure of the hyperbolicity of the traveltime-offset curve which can be used as a quantitative indicator of stacking quality.

The modelled and actual stacking velocity fields are compared by finding the stacking velocity pick that is closest to the traveltime calculated from the Gaussian-weighted regression. In order to have confidence in the inversion, the stacking velocity picks must be horizon consistent (i.e. reflections from the interpreted interfaces used to build the model must be picked during the stacking velocity analysis). Interpolation, although common, should be avoided if possible because the stacking velocity function need not vary linearly between picks. Picks that are not horizon consistent should be down-weighted during subsequent inversion.

It is essential that the forward modelling procedure correctly simulates the effect of any processing operators that have been applied to the data. Assumptions based on ideal operator behaviour are shown to not be accurate enough and it is preferable to simulate the kinematics of the actual processing operator on the modelled data. In particular, the actions of the DMO and pre-stack time-migration operators are demonstrated to be very accurately modelled by the simulation procedure and, moreover, the differences between the simulated results and the operators intended action is found to be large enough to be detected during stacking velocity analysis.

Due to the specialist nature of the problem being solved, especially the modelling of the post-processed stacking velocity field, the ray-tracing algorithms for both shooting and bending were specifically written and optimised for the inversion. This optimisation restricts the model complexity for two reasons. Firstly, the description of which interfaces the ray crosses, and in which order, must be pre-defined. This makes it difficult, although not impossible, to include complex geometry, such as multi-valued depth surfaces. Secondly, the velocity variation over a given ray segment is assumed to be linear with constant anisotropy parameters. Low frequency, non-linear variation in the velocity model can be accurately estimated by finding the best local estimate of the velocity gradient over a given ray segment.

In order for the ray-bending algorithm to be stable and fast it is important that a good initial estimate of the ray-path can be found. If the model is fully consistent with the time interpretation, the end points of the zero-offset rays from the VA location to the reflector can be accurately estimated from the connectivity relations within and between the time-migrated, the zero-offset and the depth-migrated domains. This also allows the classical problem of multiple solutions to be solved efficiently. For the modelling of the stacking velocity field, the zero-offset ray-path can be used to predict the first of the offset rays and subsequent offset rays are accurately predicted by extrapolating the differences between previous rays.

### 6.1.3 Inversion

A least-squares objective function quantitatively measures the misfit between the modelled and actual data. The objective function allows for errors in the data by weighting each contributory piece of data by its variance. By linearising the problem, a particular solution that minimizes the objective function can be found using an iterative Gauss-Newton technique. The properties of the linearised inverse problem have been studied using a singular-value decomposition. This analysis demonstrates that often there are not enough linearly independent relations between the modelled data and the model parameters to directly invert for all model parameters. A physical way of overcoming this problem is to add constraints that determine the type of model being sought by adding a term to the objective function that measures the distance from an *a priori* model. The expected deviation around the *a priori* model is defined by a covariance matrix.

The *a priori* covariance matrices are not very easy to define and are very case specific. An independent source of constraint comes from the well logs. The well information, however, is sparse, contains information on velocity variations on a much smaller scale than the seismic, and describes the region only in the immediate vicinity of the well. Due to the sparseness and the fact that the wells are usually drilled on structural highs, the parameters estimated from the logs may not be representative of the parameter values over the entire interval. Despite these difficulties, such information can still be used as an *a priori* constraint and all of the available data integrated in a rigorous way.

The choice of model parameterisation is a very important part of any inversion scheme. The method differs from conventional tomography in that the model geometry is not parameterised and inverted for. Instead, it is defined by the zero-offset time and time-dip, given by the zero-offset representation of the layer boundaries, and the current estimate of the velocity parameters. The inversion, therefore, only minimises the objective function with respect to the velocity parameters.

The input data are often insufficient to uniquely resolve all of the model parameters and there may be many combinations of model parameters that minimize the objective function. One way of measuring the non-uniqueness, or resolution, of the inverted model parameters is to estimate how the inversion scheme recovers a perturbation away from the solution. This is the basis of the resolution matrix. Each column of the resolution matrix relates to a single model parameter and contains the estimated parameter updates, required to reminimize the objective function, after a unity change of that parameter away from the solution. Interpretation of the resolution matrix is very difficult when the model parameters have different dimensions and the results are more easily interpreted if the matrix is scaled by the Hessian matrix. This normalises the model parameter changes in terms of the corresponding changes in the objective function gradient. The rescaled matrix can then be thought of in terms of the percentage of the parameter that is recovered.

As well as being insufficient, the input data are also uncertain. The *a posteriori* covariance matrix relates errors in the input data to errors in the inverted model parameters. The diagonal elements of the matrix contain the model parameter variances. The off-diagonal terms give the orientation and shape of the contours of the objective function around the solution. The covariance matrix can be used to quickly generate a number of model parameter realisations that are consistent with the error distribution of the input data.

In order to apply the inversion scheme to the problem, the partial derivatives of the modelled data values with respect to each of the model parameters must be calculated. All of the derivatives have a similar form and can be very efficiently calculated during the forward modelling process itself. Each derivative relies on calculating how the traveltime along a given ray-path is effected by a perturbation in one of the model parameters. For ease of calculation, the derivatives are calculated using a straight ray approximation, i.e. assuming a locally constant velocity field with constant anisotropy parameters.

There are two first-order effects on the traveltime along a given ray-path due to a change in the model parameters. The first is the change in traveltime along the unperturbed ray and the second is the movement of the layer boundaries to keep the zero-offset traveltime to all layer boundaries constant. The way in which a given position on the layer boundary moves due to a change in the model parameters in estimated by shooting a zero-offset ray from the point on the boundary upwards until it crosses all the intervals in which the model parameters are being changed. When inverting the model parameters within several layers, the solution becomes recursive.

## 6.2 Suggestions for Future Work

### 6.2.1 Extension of Model Parameterisation

Currently, the ray-tracing algorithms limit the complexity of the models that can be handled by the inversion software. In particular, multi-valued depth surfaces (surfaces with more than one depth at a given lateral position) are not allowed. In addition, high frequency velocity variations within a given layer interval are not taken into account accurately. In future, the ray-tracing functions could be modified to allow inversion in more structurally complex areas. These extensions will be more important is the inversion is extended to work with pre-stack data (see section 6.2.4).

In the case studies presented within this thesis, the assumption of constant anisotropy parameters within a given layer has proved sufficient to fit the stacking velocity field and tie all in the wells. In general, this may not be the case. Indeed, there is no physical reason why the anisotropy parameters should be any less spatially variant than the vertical velocity. The model parameterisation could be extended, therefore, to allow for lateral variation in the anisotropy parameters. Initial trials have shown that low frequency variations in the anisotropy parameters have been well approximated by a locally constant value over the ray segment.

The parameterisation of the anisotropy could also be extended to allow for different symmetry systems such as transverse isotropy with a horizontal axis of symmetry (TIH), transverse isotropy with a tilted axis, or more complex systems such as orthorhombic symmetry. This would allow the examination of azimuthal anisotropy (i.e. the variation of velocity with direction in a horizontal plane) which is thought to contain information about fracture orientation.

### 6.2.2 Three-Term Stacking Velocity Analysis

The stacking velocity simply describes the best hyperbolic approximation to the traveltime-offset curve for given reflection on a CMP gather. As explained in section 2.2, the moveout is generally significantly non-hyperbolic resulting in degradation of the stacked section. Several authors (for example, Castle (1988), Sena (1991), Tsvankin and Thomsen (1994), Alkhalifah and Tsvankin (1995)) have devised more complex analytical moveout equations, parameterised by three variables, that are shown to better represent the reflection moveout.

During processing, the optimal parameters can still be found by a semblance analysis method, although this must be extended into a higher order space which causes significant practical difficulty. Also, the semblance widths on higher order terms tend to be extremely large and therefore picks have large error bars. It is hoped that with long offset data, however, higher-order velocity analysis will prove fruitful. During inversion, the modelled moveout parameters can be calculated easily as the Gaussian-weighted regression can be adapted to work for any analytical moveout expression.

### 6.2.3 Multi-Component Seismic Inversion

Recently, multi-component seismic surveys, in which shear waves (S) are recorded in addition to compressional waves (P), have generated a great deal of interest within the oil industry. It is hoped that such surveys will improve both structural imaging and reservoir characterisation.

During the processing of these data, a stacking velocity analysis can be carried out on the P-S mode converted arrivals as well as those of the P-P data. These velocity analyses could then be inverted at the same time as the P-P wave velocity and could help improve resolution of the velocity parameters, especially the anisotropy. This requires some further work to simulate different processing operators, such as the common-conversion point DMO operator during the modelling of the mode converted stacking velocity which is routinely applied to multi-component data.

### 6.2.4 Extension to Pre-Stack Inversion

It has long been recognised that pre-stack depth-migration itself is a powerful tool for validating and building velocity models: the redundancy present in pre-stack data can be exploited via the requirement for the consistency of images produced by individual migration of different common-offset or common-shot gathers. Therefore the analysis of gathers of traces from the migrated volume which correspond to the same lateral position (but different offsets or shots), known as common image gathers (CIG), yields information about inaccuracies in the current velocity model. If the velocity model is correct, the reflections will appear at the same depths in each trace of the gather. Residual moveout on the gather is evidence of incorrect velocities.

Existing methods of pre-stack velocity analysis can be separated into two main classes. Firstly, migration velocity analysis (e.g. Faye and Jeannot, 1986, Audebert and Diet, 1996) techniques are usually based more or less explicitly upon scanning over a range of velocities and picking the one that produces the flattest CIGs for reflectors of interest. Such approaches can integrate the migration and scanning/focusing operations, but may nonetheless be inefficient when several parameters must be estimated simultaneously, e.g. for anisotropy and lateral variation. The second class proceeds by iterative tomographic inversion of residuals from CIGs after pre-stack depth-migration (e.g. Stork, 1992, Kosloff et al., 1996). Despite the possible cost of picking and iterative migration, this route seems likely to be more effective for complex cases.

The model built during the inversion can be used to create sets of CIGs which differ slightly from those described above: at a selected lateral position, a fan of rays can be traced through the model, with the appropriate range of offsets, for each interface, such that all the rays reflect at that lateral position. After shooting a zero-offset ray, this is achieved by an efficient bending algorithm in which successive rays are initialised using the previous ones in a kind of offset continuation procedure. Each ray in a gather is characterised by a traveltime and its source and receiver positions. Given this information, small sections of the corresponding seismic trace, centred around the modelled traveltimes, can be extracted directly from the seismic data volume to create gathers associated with each interface at the chosen lateral position. As the model is consistent with the initial interpretation, the interpreted reflection event close to the centre of the extracted trace segments at near offset. This can be used to help identify the phase of the seismic wavelet associated with each reflector.

For the correct velocity model, the central event in the gather will appear flat across all offsets because of the effective subtraction of the ray-trace moveout. These model-based image gathers can be used to validate the velocity model and calculate velocity model updates in places where the gathers are not flat. The traveltime residuals for non-zero offsets can either be picked manually or estimated automatically by correlation or analysis similar to that suggested by Kosloff et al. (1996). Note that even with a perfect velocity model, only the central part of each gather will appear flat as only the data around the ray-trace traveltime is migrated correctly. The inversion scheme presented in this thesis could be modified to update the interval velocity model in order to minimise the traveltime residuals (Sexton and Williamson, 1998). The use of pre-stack data should help to reduce non-uniqueness of parameter estimates, particularly for the anisotropy.

# 7.0 References

Aki, K. and Richards, P.G., 1980, Quantitative Seismology, Theory and Methods. W.H. Freeman and Company (2 volumes).

Al-Chalabi, M., 1994, Seismic velocities - a critique. First Break, **12**, 589-596.

Alkhalifah, T. and Tsvankin, I., 1995, Velocity analysis for transversely isotropic media, Geophysics, **60**, 1550-1566.

Audebert, F. and Diet, J.P., 1996, CRP-scans from 3D pre-stack depth migration: a new tool for velocity model building. Abstracts of the 58th EAGE Conference, Amsterdam.

Backus, G.E. and Gilbert, J.F., 1968, The resolving power of gross earth data. Geophys. J. Roy. Astr. Soc., **16**, 169-205.

Ball, G., 1995, Estimation of anisotropy and anisotropic 3-D depth migration, offshore Zaire. Geophysics, **60**, 1495-1513.

Banik, N.C., 1984, Velocity anisotropy of shales and depth estimation in the North Sea basin. Geophysics, **49**, 1411-1419.

Berkhout, A.J., 1980, Seismic migration. Elsevier Science Publ. Co.

Berryman, J.G., 1979, Long-wave elastic anisotropy in tranversely isotropic media. Geophysics, **44**, 896-917.

Bevc, D., Black, J.L. and Palacharla, G., 1995, Plumes: response of time migration to lateral velocity variation. Geophysics, **60**, 1118-1127.

Castle, R.J., 1988, Shifted hyperbolas and normal moveout. Extended abstracts of the 58th SEG meeting, 894-896.

Cerveny, V., 1972, Seismic rays and ray intensities in inhomogeneous anisotropic media. Geophys. J. Roy. Astr. Soc., **29**, 1-13.

Chiu, S.K. and Stewart, R., 1987, Tomographic determination of three-dimensional seismic velocity structures using wells, vertical seismic profiles and surface seismic data. Geophysics, **52**, 1058-1098.

Claerbout, J.F., 1970, Coarse grid calculations of waves in inhomogeneous media with application to delineation of complicated seismic structure. Geophysics, **35**, 407-418.

Claerbout, J.F., 1971, Towards a unified theory of reflector mapping. Geophysics, **36**, 467-481.

Deregowski, D. and Rocca, F., 1981, Geometrical optics and wave theory for constant-offset sections in layered media. Geophysical Prospecting, **29**, 374-387.

Deregowski, S.M., 1986, What is DMO? First Break, **4(7)**, 7-24.

Dix, C.H. 1955. Seismic velocities from surface measurements. Geophysics, **20**, 68-86.

Faye, J.P. and Jeannot, J.P., 1986, Prestack migration velocities from focussing depth analysis: Expanded abstracts of the 56th SEG meeting, Houston, 438-440.

Feynmann, R.P, Leighton, R.B. and Sands, M.L., 1964, The Feynmann lectures on Physics (3 volumes). Addison-Wesley Publishing Co., Inc., Reading, Massachusetts.

Hadley, D.M., Thorson, J. and Maher, S., 1988., Increasing interpretation accuracy: A new approach to interval velocity estimation. The Leading Edge, **7**, 13-16.

Hagedøørn, J.G., 1954, A process of seismic reflection interpretation. Geophysical Prospecting, **7**, 158-182.

Hale, D., 1991, Dip move out processing: course note series. Volume 4 SEG, SN. Domenico editor.

Hatton, L., Worthington, M.H. and Makin, J., 1986, Seismic data processing. Blackwell Sci. Publns.

Hubral P., 1977, Time migration - some ray theoretical aspects. Geophysical Prospecting, **25**, 738-745.

Iversen, E. and Gjøystdal H., 1996, Event-orientated velocity estimation based on prestack data in time or depth domain. Geophysical Prospecting, **44**, 643-686.

Jackson, D.D., 1972, Interpretation of inaccurate, insufficient and inconsistent data. Geophysical J.Roy. Astr. Soc., **28**, 97-109.

Judson, D.R., Schulz, P.S. and Sherwood, J.W.C., 1978, Equalising the stacking velocities of dipping events via DEVILISH. Extended abstracts of the 48th SEG convention, San Fransisco.

Kaarsberg, E.A., 1959, Introductory studies of natural and artificial argillaceous aggregates by sound propagation and X-ray diffraction methods. J. Geol., **67**, 447.

Khare V., 1991, A new ray-theoretic interpretation of time migration. Expanded abstracts of 61st SEG, 1492-1496.

Kosloff, D., Sherwood, J., Koren, Z., Machet, E. and Falkovitz, Y., 1996, Velocity and interface depth determination by tomography of depth migrated gathers. Geophysics, **61**, 1511-1523.

Lanczos, C., 1961, Linear differential operators. Van Nostrand, Princeton, NJ.

Levenberg, K., 1944. A method for the solution of certain non-linear problems in least-squares. Quart. appl. math., **2**, 162-168.

Levin, F.K., 1971, Apparent velocity from dipping interface reflections. Geophysics, **36**, 510-516.

Levin, F.K., 1990, Reflection from a dipping plane - Transversely isotropic solid. Geophysics, **55**, 851-855.

Lynn, W., Gonzalez, A. and Mackay, S., 1991, Where are the fault-plane reflections? Expanded abstracts of 61st SEG, 1151-1154.

Marcoux, M.D., Godfrey, R.J. and Notfors, C.D.B., 1987, Migration for optimum velocity evaluation and stacking (MOVES). Abstracts of the 49th EAEG conference, Belgrade.

Neidell, N.S. and Taner, M.T., 1971, Semblance and other coherency measures for multichannel data. Geophysics, **36**, 482-97.

Nye, J.F., 1957, Physical properties of crystals. Oxford Press.

Oliver, D.S., 1995, Moving averages for gaussian simulation in two and three dimensions. Mathematical Geology, **27**, 8, 939-960.

Raynaud, B. and Thore, P., 1993, Real time migration operators simulated by anisotropic ray tracing. Abstracts of the 55th EAEG meeting, Stavanger.

Robein, E., Raynaud, B. and Sexton P., 1995, Accurate simulation of the kinematics of DMO in travel-time inversion. Abstracts of the 57th EAGE meeting, Glasgow.

Robein, E., Sexton P. and Philandrianos, F., 1997, The significance of stacking velocities after 3D prestack time migration. Abstracts of the 59th EAGE meeting, Geneva.

Sena, A.G., 1991, Seismic traveltime equations for azimuthally anisotropic and isotropic media: estimation of interval elastic properties, Geophysics, **56**, 2090-2101.

Sexton, P. and Robein, E., 1996, Different simulations of the kinematics of time-migration - a review. Abstracts of the 58th EAGE meeting, Amsterdam.

Sexton, P. and Williamson, P., 1998, 3D anisotropic velocity estimation by model-based inversion of pre-stack traveltimes. Expanded abstracts of the 68th meeting of the SEG (submitted), New Orleans.

Shearer, P.M. and Chapman C.H., 1988, Ray tracing in anisotropic media with a linear gradient. Geophysical Journal, **94**, 575-580.

Stork, C. and Clayton, R.W., 1991, Linear Aspects of tomographic velocity analysis. Geophysics, **56**, 483-495.

Stork, C., 1992, Reflection tomography in the postmigrated domain. Geophysics, **57**, 680-692.

Tarantola, A. and Valette, B., 1982, Inverse problems = quest for information, J. Geophys., **50**, 159-170.

Thomsen, L., 1986, Weak elastic anisotropy. Geophysics, **51**,1954-1966.

Thore, P.D., de Bazelaire, E. and Ray, M.P., 1994, The three-parameter equation: An efficient tool the enhance the stack. Geophysics, **59**, 297-308.

Tsvankin, I. and Thomsen, L., 1994, Nonhyperbolic reflection moveout in anisotropic media, Geophysics, **59**, 1290-1304.

Whitcombe, D.N., 1994, Fast model building using demigration and single-step ray migration, Geophysics, **59**, 439-449.

Williamson, P.R. and Raynaud, B., 1995, Approximations for fast ray-tracing through layered anisotropic media with velocity gradients, Abstracts of the 57th EAEG conference, Glasgow.

Williamson, P.R. Sexton, P. and Xu, S., 1997, Integrating observations of elastic anisotropy: constrained inversion of seismic kinematic data, Expanded abstracts of the 67th meeting of the SEG, 1695-1698.

Xu, S. and White, R.E., 1996, A physical model for shear-wave velocity prediction: Geophysical Prospecting, **43**, 91-118.