

Time-migration velocity estimation using Fréchet derivatives based on nonlinear kinematic migration/demigration solvers

HAO ZHAO¹, ANDERS UELAND WALDELAND², DANY RUEDA SERRANO³, MARTIN TYGEL³
AND EINAR IVERSEN⁴

- 1 Department of Geosciences, University of Oslo, P.O. Box 1047 Blindern, N-0316 Oslo, Norway (zhaohao7109@gmail.com)
- 2 Department of Informatics, University of Oslo, P.O. Box 1080 Blindern, N-0316 Oslo, Norway
- 3 Center for Petroleum Studies, Rua Cora Coralina 350, Cidade Universitária, 13083-896 Campinas - SP, Brazil
- 4 Department of Earth Science, University of Bergen, P.O. Box 7803, N-5020 Bergen, Norway

Received: September 13, 2019; Revised: November 10, 2019; Accepted: November 18, 2019

ABSTRACT

Advanced seismic imaging and inversion are dependent on a velocity model that is sufficiently accurate to render reliable and meaningful results. For that reason, methods for extracting such velocity models from seismic data are always in high demand and are topics of active investigation. Velocity models can be obtained from both the time and depth domains. Relying on the former, time migration is an inexpensive, quick and robust process. In spite of its limitations, especially in the case of complex geologies, time migration can, in many instances (e.g. simple to moderate geological structures), produce image results compatible to the those required for the project at hand. An accurate time-velocity model can be of great use in the construction of an initial depth-velocity model, from which a high-quality depth image can be produced. Based on available explicit and analytical expressions that relate the kinematic attributes (namely, traveltimes and local slopes) of local events in the recording (demigration) and migrated domains, we revisit tomographic methodologies for velocity-model building, with a specific focus on the time domain, and on those that makes use of local slopes, as well as traveltimes, as key attributes for imaging. We also adopt the strategy of estimating local inclinations in the time-migrated domain (where we have less noise and better focus) and use demigration to estimate those inclinations in the recording domain. On the theoretical side, the main contributions of this work are twofold: 1) we base the velocity model estimation on kinematic migration/demigration techniques that are nonlinear (and therefore more accurate than simplistic linear approaches) and 2) the corresponding Fréchet derivatives take into account that the velocity model is laterally heterogeneous. In addition to

providing the comprehensive mathematical algorithms involved, three proof-of-concept numerical examples are demonstrated, which confirm the potential of our methodology.

Keywords: migration velocity estimation, kinematic time migration and demigration, kinematic attributes

1. INTRODUCTION

In seismic data, wave responses to subsurface structures (e.g. layer interfaces or geological bodies) manifest themselves in the form of signal alignments, referred to as seismic events. In the simplest case of a 2D section, such as shot or common-midpoint (CMP) gathers, data points along an event occupy a traveltimes strip around a curve within that section. In the case of 2D acquisition, with the arbitrary location of sources and receivers, an event aligns on a strip around a surface in a 3D data volume. For 3D data acquisition, the alignment occurs in the neighbourhood of a hypersurface within a 5D data volume. Seismic events in the data set represent the time link for accessing information on depth geological structures. For complex geologies and noisy environments, however such events, are generally obscured or hidden within the data, requiring proper processing schemes for their identification and use.

As recognised in the literature, it is advantageous to describe an event alignment by means of slopes and curvatures that pertain to their individual points. Referred to as local kinematic parameters of an event point, such quantities are able to provide parametric traveltimes approximations of the seismic event in the vicinity of that original event point (see, e.g., Ursin, 1982; Hubral, 1999; Landa *et al.*, 1999, 2010; Perroud *et al.*, 1999; Jäger *et al.*, 2001; Bergler *et al.*, 2002; Hertweck *et al.*, 2007; Landa, 2007; Tygel and Santos, 2007; Berkovitch *et al.*, 2008; Fomel and Kazinnik, 2012; Bloot *et al.*, 2018; Gelius and Tygel, 2015). In this way, the search for seismic events is reduced to the search for event points and their corresponding local kinematic parameters. Each of the traveltimes is associated with specific events (e.g., primary reflections or diffractions) for which, conceptually, best approximations are expected. As also documented in the literature, the formulation outlined above allows for the use of coherence (e.g. semblance) analysis (see, e.g., Neidell and Taner, 1971) for the detection of the desired information (see, e.g., Bonomi *et al.*, 2009; Minato *et al.*, 2012; Barros *et al.*, 2015; Garabito *et al.*, 2017; Walda and Gajewski, 2017; Waldeland *et al.*, 2019). Local kinematic parameters have shown to be rather instrumental in a number of applications of seismic processing, imaging and inversion. These include a) poststack/prestack data enhancement (Baykulov and Gajewski, 2009; Faccipieri *et al.*, 2016); b) diffraction separation and imaging (Bauer *et al.*, 2009; Berkovitch *et al.*, 2009; Klovov and Fomel, 2012; Faccipieri *et al.*, 2013; Rad *et al.*, 2018); c) time migration/demigration (Spinner and Mann, 2006; Bóna, 2011; Dell and Gajewski, 2011; Iversen *et al.*, 2012; Coimbra *et al.*, 2013, 2016a; Dell *et al.*, 2014); d) data interpolation and regularisation (Hoecht *et al.*, 2009; Coimbra *et al.*, 2016b); e) attribute-oriented seismic processing (Fomel, 2002, 2007a,b; Zhang *et al.*, 2002; Cooke *et al.*, 2009; Stovas and Fomel, 2015; Khoshnavaz *et al.*, 2016a,b; Khoshnavaz, 2017; Vanelle *et al.*, 2018; Yuan *et al.*, 2018, 2019).

In addition to the applications mentioned above, local kinematic parameters have made a substantial impact on ray-based reflection tomography through the introduction of new algorithms that are able to use slope parameters in addition to the traveltimes normally used in conventional schemes. In the classic literature, such an approach is referred to as slope tomography (Rieber, 1936; Riabinkin, 1957; Sword, 1986). A powerful extension of slope tomography was proposed by Billette and Lambaré (1998) under the name of stereotomography. Such an extension takes full advantage of the more recently available theoretical and practical (computational) advances in inverse-problem solving. To overcome the difficulties of traveltimes and slope picking of local coherent events in the recording time domain, Guillaume et al. (2001) and Chauris et al. (2002a,b) proposed that the picking should first be done in the prestack depth migration (PSDM), and then demigrated back to the recorded domain. Such an approach demonstrates the interest in using kinematic migration and its counterpart kinematic demigration for velocity-model building. In fact, kinematic migration and kinematic demigration also have a history in the seismic literature, having been used to map the local kinematic parameters (traveltimes, slopes and curvatures) from/to the recording time domain to/from the migrated time or depth domain (Shah, 1973; Kleyn, 1977; Hubral and Krey, 1980; Gjøystdal and Ursin, 1981; Ursin, 1982; Iversen and Gjøystdal, 1996; Iversen, 2004; Douma and de Hoop, 2006; Stolk et al., 2009).

Concerning kinematic demigration, Whitcombe et al. (1994) introduced a zero-offset kinematic time demigration scheme using the constant migration velocity assumption. Söllner et al. (2004) and Söllner and Andersen (2005) further investigated zero-offset kinematic migration and demigration under the framework of ray theory. A more systematic generalisation and extension of kinematic time migration and demigration was developed by Iversen et al. (2012). Their proposed technique extends the kinematic time migration and demigration from a zero-offset to a finite-offset, including both first-order (slope) and second-order (curvature) travel time derivatives during the mapping, and taking into account the diffraction time functions of the second order in the common-image gather location, the source-receiver offset and the migration aperture. To use kinematic migration and demigration in velocity estimation, another type of tomography technique, nonlinear tomography, has been proposed by different authors. Adler et al. (2008) proposed a technique of nonlinear 3D tomographic inversion to replace the conventional linear process of PSDM using residual moveout analysis (RMO). Guillaume et al. (2008) addressed the flexibility of using the kinematic invariants (traveltimes and slopes in the unmigrated recording domain) for nonlinear depth tomography. Lambaré et al. (2009) extended this nonlinear inversion concept to time imaging. These proposed non-linear tomographic approaches greatly reduce the turnaround times and cost of the PSDM and velocity estimation by replacing the conventional PSDM-plus-velocity-RMO analysis with the kinematic demigration and migration internal velocity updating scheme. Due to its mentioned efficiency and effectiveness, the non-linear tomography technique attracted much more attention and development through industrial application and academic research in recent years (Guillaume et al., 2013; Messud et al., 2015).

In this paper, we revisit the kinematic time migration and demigration scheme (Iversen et al., 2012), developing a detailed numerical scheme for time-migration velocity estimation. As the authors are aware, a general description of time-migration velocity estimation using nonlinear mapping processes, based on kinematic time migration and

demigration, has been missing from the geophysical theory. Considering the importance of the generalisation of time-migration velocity estimation using Fréchet derivatives based on nonlinear kinematic migration and demigration solvers, this paper provides the missing element for event-oriented seismic velocity estimation. In the following, we first describe the diffraction time function for time migration, the time-migration velocity model. Thereafter, we review the algorithms for kinematic time migration and demigration, and derive the Fréchet derivatives using sensitivity analysis of the kinematic time migration with respect to the time-migration velocity model. Finally, we formulate the linear system for time-migration velocity estimation. Based on the derived mathematical algorithms, we provide three proof-of-concept examples. First, we verify the derived formulation for kinematic time migration and the calculation of the Fréchet derivatives based on a constant time-migration velocity model. Next, we present an example of a time-migration velocity estimation based on a 2D synthetic dataset. In the last - a 3D real-data example, in which we did not have access to the individual, non-stacked, time-migrated common-offset panels - we use a workaround by identifying the times and slopes on a stacked image in the time-migration domain, and verifying the availability of the 3D time-migration velocity estimation based on the theoretical formulations. The three examples confirm the potential of our methodology.

2. DIFFRACTION TIME FUNCTION FOR TIME MIGRATION

We describe the diffraction time function for time migration and the coordinates and coefficients related to it. The description follows *Iversen et al. (2012)* but includes an extension from second to fourth order.

2.1. Coordinates

We use a 2D Cartesian coordinate system (ξ_1, ξ_2) for describing the acquisition geometry of 3D seismic experiments. As shown in Fig. 1, in the horizontal measurement plane, consider a source point, $\mathbf{s} = (s_1, s_2)$, and a receiver point, $\mathbf{r} = (r_1, r_2)$. The midpoint \mathbf{x} and half-offset \mathbf{h} are given as

$$\mathbf{x} = \frac{1}{2}(\mathbf{r} + \mathbf{s}), \quad \mathbf{h} = \frac{1}{2}(\mathbf{r} - \mathbf{s}). \quad (1)$$

For the output point of time migration, i.e., the image-gather location, we use the notation \mathbf{m} . The aperture vector \mathbf{a} is defined as

$$\mathbf{a} = \mathbf{x} - \mathbf{m}. \quad (2)$$

We will also need the quantities \mathbf{h}^S and \mathbf{h}^R ,

$$\mathbf{h}^S = \mathbf{s} - \mathbf{m} = \mathbf{a} - \mathbf{h}, \quad \mathbf{h}^R = \mathbf{r} - \mathbf{m} = \mathbf{a} + \mathbf{h}, \quad (3)$$

referred to, respectively, as the source-offset vector and the receiver-offset vector.

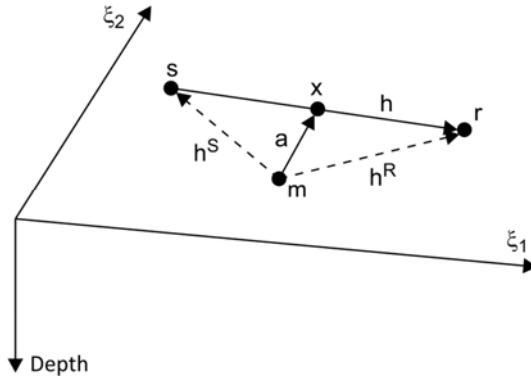


Fig. 1. Coordinate system used for describing the 3D seismic experiments. Source (s), receiver (r), common midpoint (x) and common image point (m) are defined in the horizontal measurement plane. The vectors of aperture (a), half-offset (h), source-offset (h^S) and receiver-offset (h^R) are outlined.

2.2. Diffraction time

Assuming single scattering and traveltimes fields without caustics, the diffraction time function for time migration, T^D , can be introduced as the traveltimes from the source point to the scattering point, T^S , plus the traveltimes from the scattering point to the receiver, T^R ,

$$t = T^D(\mathbf{h}, \mathbf{a}, \mathbf{m}, \tau) = T^S(\mathbf{h}^S, \mathbf{m}, \tau) = T^R(\mathbf{h}^R, \mathbf{m}, \tau). \quad (3)$$

In Eq. (3) the source- and receiver-offset vectors are expressed in terms of \mathbf{a} and \mathbf{h} . We note that $t = t^D$ is the (two-way) diffraction time from source to receiver, while τ is the migration time, i.e., the time coordinate of the domain of the time-migrated data. The diffraction time function T^D is required stationary for $\mathbf{h} = \mathbf{a} = 0$.

Let the source and receiver traveltimes correspond to the same wave type, e.g., a direct P wave. We establish a Taylor expansion for the squared one-way time T^{S^2} ,

$$\begin{aligned} T^{S^2} = & \left(\frac{\tau}{2}\right)^2 + \frac{1}{2} \frac{\partial^2 (T^S)^2}{\partial h_I^S \partial h_J^S} h_I^S h_J^S + \frac{1}{6} \frac{\partial^3 (T^S)^2}{\partial h_I^S \partial h_J^S \partial h_K^S} h_I^S h_J^S h_K^S \\ & + \frac{1}{24} \frac{\partial^4 (T^S)^2}{\partial h_I^S \partial h_J^S \partial h_K^S \partial h_L^S} h_I^S h_J^S h_K^S h_L^S + \dots, \end{aligned} \quad (5)$$

and an analogous expansion for T^{R^2} . The first order derivatives of T^{S^2} and T^{R^2} are zero, as the diffraction time function is stationary for $\mathbf{h} = \mathbf{a} = 0$. Since the source and

receiver traveltime fields have the same wave type, the Taylor coefficients in the expansions of T^{S^2} and T^{R^2} must be equal. Hence, we can write

$$\begin{aligned} T^{S^2} &= \left(\frac{\tau}{2}\right)^2 + S_{IJ}^{M(2)} h_I^S h_J^S + S_{IJK}^{M(3)} h_I^S h_J^S h_K^S + S_{IJKL}^{M(4)} h_I^S h_J^S h_K^S h_L^S + \dots, \\ T^{R^2} &= \left(\frac{\tau}{2}\right)^2 + S_{IJ}^{M(2)} h_I^R h_J^R + S_{IJK}^{M(3)} h_I^R h_J^R h_K^R + S_{IJKL}^{M(4)} h_I^R h_J^R h_K^R h_L^R + \dots. \end{aligned} \quad (6)$$

If the lateral variations within the medium are small, and anisotropy effects are laterally symmetric, the odd terms in Eq. (6) can be neglected. Applying this assumption and truncating the series after the fourth order terms, our diffraction time function is given by Eq. (4) with arguments given by Eq. (3) and source/receiver time fields

$$\begin{aligned} T^S &= \left[\left(\frac{\tau}{2}\right)^2 + S_{IJ}^{M(2)} h_I^S h_J^S + S_{IJKL}^{M(4)} h_I^S h_J^S h_K^S h_L^S \right]^{1/2}, \\ T^R &= \left[\left(\frac{\tau}{2}\right)^2 + S_{IJ}^{M(2)} h_I^R h_J^R + S_{IJKL}^{M(4)} h_I^R h_J^R h_K^R h_L^R \right]^{1/2}. \end{aligned} \quad (7)$$

For the subtle details on the partial derivatives of the diffraction-time function, see Appendix A.

2.3. Diffraction time coefficients

The coefficients in Eq. (7) may be expressed, for example, as

$$S_{IJ}^{M(2)} = 2\tau \frac{\partial^2 T^S}{\partial a_I \partial a_J}, \quad (8)$$

$$\begin{aligned} S_{IJKL}^{M(4)} &= \frac{1}{12} \left(2\tau \frac{\partial^4 T^S}{\partial a_I \partial a_J \partial a_K \partial a_L} \right. \\ &\quad \left. + \frac{\partial^2 T^S}{\partial a_I \partial a_J} \frac{\partial^2 T^S}{\partial a_K \partial a_L} + \frac{\partial^2 T^S}{\partial a_I \partial a_K} \frac{\partial^2 T^S}{\partial a_J \partial a_L} + \frac{\partial^2 T^S}{\partial a_I \partial a_L} \frac{\partial^2 T^S}{\partial a_J \partial a_K} \right), \end{aligned} \quad (9)$$

where all derivatives are taken for $\mathbf{a} = \mathbf{h} = 0$. Observe, however, that the coefficients $S_{IJ}^{M(2)}$ and $S_{IJKL}^{M(4)}$ are functions of \mathbf{m} and τ . We can relate these coefficients to derivatives of the diffraction time function and its square, as follows,

$$\frac{\partial^2 T^D}{\partial a_I \partial a_J} = \frac{4}{\tau} S_{IJ}^{M(2)}, \quad (10)$$

$$\frac{\partial^4 T^D}{\partial a_I \partial a_J \partial a_K \partial a_L} = \frac{16}{\tau} \left[3S_{IJKL}^{M(4)} - \frac{1}{\tau^2} \left(S_{IJ}^{M(2)} S_{KL}^{M(2)} + S_{IK}^{M(2)} S_{JL}^{M(2)} + S_{IL}^{M(2)} S_{JK}^{M(2)} \right) \right], \quad (11)$$

$$\frac{\partial^2 \left(T^D \right)^2}{\partial a_I \partial a_J} = 8S_{IJ}^{M(2)}, \quad (12)$$

$$\frac{\partial^4 \left(T^D \right)^2}{\partial a_I \partial a_J \partial a_K \partial a_L} = 96S_{IJKL}^{M(4)}. \quad (13)$$

2.4. A simplified setup of diffraction time coefficients

Equation (7) is formulated to allow for a full coverage of directions and magnitudes of the source- and receiver-offset vectors. For situations with limited coverage it is however useful to have a simplified setup for the diffraction time coefficients.

One approach often used is to assume that the diffraction time moveout is rotationally symmetric. We then write

$$S_{IJ}^{M(2)} = S^{M(2)} \delta_{IJ}, \quad S_{IJKL}^{M(4)} = S^{M(4)} \delta_{IJ} \delta_{KL}. \quad (14)$$

where $S^{M(2)}$ and $S^{M(4)}$ are scalar functions of the coordinates (\mathbf{m}, τ) . Substituting Eq. (14) in Eq. (7) yields

$$T^S = \left[\left(\frac{\tau}{2} \right)^2 + S^{M(2)} \left(h^S \right)^2 + S^{M(4)} \left(h^S \right)^4 \right]^{1/2}, \quad (15)$$

$$T^R = \left[\left(\frac{\tau}{2} \right)^2 + S^{M(2)} \left(h^R \right)^2 + S^{M(4)} \left(h^R \right)^4 \right]^{1/2}.$$

Here, the quantities h^S and h^R denote the magnitudes of the respective source- and receiver-offset vectors, \mathbf{h}^S and \mathbf{h}^R .

2.5. Time-migration velocity model

The time-migration velocity model is defined on a three-dimensional rectangular grid in the variables $\xi_1 = m_1$, $\xi_2 = m_2$, $\xi_3 = \tau$, where we add ξ_3 to describe the velocity model in the time domain. Model parameters related to cells or vertices within the grid will be unknowns in a velocity estimation process.

Our velocity model is described in terms of a multi-component vector function, M_λ , $\lambda = 1, \dots, N^\lambda$, where each component function $M_\lambda(\xi_1, \xi_2, \xi_3)$ corresponds to one of the coefficients of the diffraction time function. For different contexts for these coefficients,

see Eqs (6), (7), and (15). The vector function M_λ is introduced to yield a compact equivalent of the diffraction time coefficients under consideration. For example, for the simplified situation of Eq. (15) we take

$$M_1 = S^{M(2)}, \quad M_2 = S^{M(4)}. \quad (16)$$

To allow the differentiability up to the second order, we use a local cubic spline function (see Appendix C) to describe the velocity model. Bicubic and tricubic spline functions are used for 2D and 3D grids respectively. In Appendix E, we give a detailed description of the parameterization of the time-migration velocity model.

3. KINEMATIC TIME MIGRATION AND DEMIGRATION

Kinematic time-migration maps the local kinematic parameters (traveltimes, slopes and curvatures) from the recording time domain of seismic data to the time-migration domain. The inverse process, kinematic time demigration, maps the local kinematic parameters in time-migration domain back to the recording time domain. As illustrated in Fig. 2, by using the diffraction function with its associated partial derivatives, and a known time-migration velocity model, kinematic time migration and demigration are able to map the local kinematic parameters between the recording time domain and the migration time domain. In this paper, we focus on the utilization of the kinematic parameters up to the first order for time-migration velocity estimation. For mapping of second order kinematic parameters, see *Iversen et al. (2012)*.

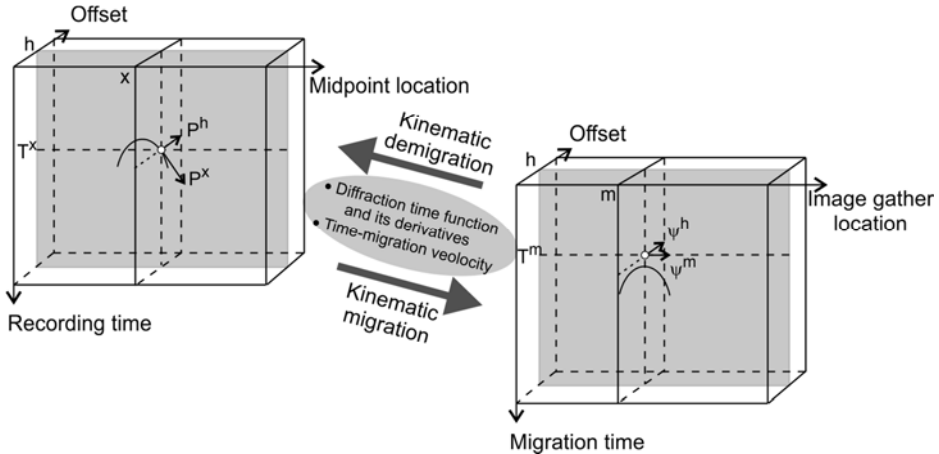


Fig. 2. Schematic overview of kinematic time migration and demigration for 2D prestack seismic data set. Based on the known diffraction time function with its associated derivatives, and a time-migration velocity model, the local kinematic parameters (x , T^x , p^x , p^h) in recording domain can be forward/backward mapped to/from the counterpart (m , T^m , ψ^m , ψ^h) in migration domain by kinematic time migration/kinematic time demigration.

The local kinematic parameters can be extracted from both the recording domain and the migration domain. Since the time-migrated dataset in general contains less noise and is more easily interpreted, the kinematic parameters extraction is commonly often applied in the migration domain, and subsequently demigrated to the recording domain. The time-migrated dataset is normally generated by prestack time migration of the recorded seismic data with an initial time-migration velocity model. The migrated dataset is not necessarily to be optimal but is assumed to have a good resolution for the local kinematic parameters picking. Once the picking process applied in the migration domain, the local kinematic parameters are able to be mapped to the recording domain by kinematic demigration with the same initial velocity model. This process needs the computation of the first and second order partial derivatives of the known diffraction-time function (i.e. Eqs (4), (7), and (15)), which are then used in the kinematic demigration and migration for velocity estimation. We give the formulation of kinematic time demigration and migration in the following context.

3.1. Algorithm for kinematic time demigration

Consider Eq. (43) in *Iversen et al. (2012)*, referred to as the *consistency equation*,

$$\frac{\partial T^D}{\partial a_I} - \frac{\partial T^D}{\partial m_I} = \frac{\partial T^D}{\partial \tau} \frac{\partial \tau}{\partial m_I}, \quad (17)$$

with $I = 1, 2$. This equation arises from the fact that the reflection time response in the recording domain can be interpreted as an envelope of a continuous ensemble of elementary diffraction time responses. In this perspective, the reflection time must equal the elementary diffraction time at the location under consideration, and the slopes of the two time functions must also be equal at that point.

We can use Eq. (17) to obtain the aperture vector corresponding to a kinematic time demigration,

$$\hat{a}_I = \hat{x}_I - m_I. \quad (18)$$

Under special conditions the vector $\hat{\mathbf{a}} = (\hat{a}_I)$ can be obtained analytically, see Eq. (78) in *Iversen et al. (2012)*.

Based on Eq. (17) we introduce a vector function $f_I(\mathbf{a})$ so that

$$f_I(\mathbf{a}) = \frac{\partial T^D}{\partial a_I} - \frac{\partial T^D}{\partial m_I} - \frac{\partial T^D}{\partial \tau} \frac{\partial \tau}{\partial m_I}, \quad (19)$$

where $\mathbf{a} = \hat{\mathbf{a}}$ is the sought unknown aperture vector. As a consequence, the system of equations to be solved has the general form

$$f_I(\mathbf{a}) = 0. \quad (20)$$

One possible approach to obtain a solution is to use the Newton-Raphson iteration method. Let us assume that the component values of the function f_I is known for some starting component values $\mathbf{a} = \mathbf{a}^{(0)}$. For these starting values we compute the first-order partial derivatives of f_I ,

$$\frac{\partial f_I}{\partial a_J}(\mathbf{a}^{(0)}) = \frac{\partial^2 T^D}{\partial a_I \partial a_J} - \frac{\partial^2 T^D}{\partial m_I \partial a_J} - \frac{\partial^2 T^D}{\partial \tau \partial a_J} \frac{\partial \tau}{\partial m_I}, \quad (21)$$

with all the partial derivatives on the right-hand side evaluated for $\mathbf{a} = \mathbf{a}^{(0)}$. To first order we can then write

$$0 - f_I(\mathbf{a}^{(0)}) = \frac{\partial f_I}{\partial a_J}(\mathbf{a}^{(0)}) (a_J^{(1)} - a_J^{(0)}). \quad (22)$$

Defining

$$W_{IJ}(\mathbf{a}^{(0)}) = \frac{\partial f_I}{\partial a_J}(\mathbf{a}^{(0)}), \quad \bar{W}_{MN}(\mathbf{a}^{(0)}) = \left\{ \left\{ \frac{\partial f_I}{\partial a_J}(\mathbf{a}^{(0)}) \right\}^{-1} \right\}_{MN}, \quad (23)$$

equation (22) yields

$$a_J^{(1)} = a_J^{(0)} - \bar{W}_{JL}(\mathbf{a}^{(0)}) f_L(\mathbf{a}^{(0)}). \quad (24)$$

Equation (24) provides a first-order update of the aperture vector. We can now compute the components $f_I(\mathbf{a}^{(1)})$ and check if these are sufficiently close to zero. Should that be the case, the Newton-Raphson process is completed and has returned the solution $\hat{\mathbf{a}} = \mathbf{a}^{(1)}$. Otherwise, the process goes on to the next iteration. The process is stopped when both components f_I are sufficiently close to zero. It is also stopped if we, in rare cases, do not attain convergence to a meaningful solution. Recall that the process is in general non-linear, so it may, in principle, converge to a non-wanted solution. For our purposes, however, the process has proven very robust with respect to the choice of starting values.

Equations (19) and (21) contain first- and second-order partial derivatives of the diffraction-time function, T^D . For the calculation of such derivatives, see Appendices C and D in *Iversen et al. (2012)*.

Now that we know the kinematic demigration aperture vector, $\hat{\mathbf{a}} = (\hat{a}_I)$, it is easy to find the time T^X and the slopes $p_I^x = \partial T / \partial x_I$ and $p_I^h = \partial T / \partial h_I$ resulting from the kinematic demigration. To compute T^X and p_I^x , we simply use the equalities

$$T^X = T^D(\mathbf{h}, \hat{\mathbf{a}}, \mathbf{m}, \tau^M), \quad p_I^x = \frac{\partial T^D}{\partial a_I}(\mathbf{h}, \hat{\mathbf{a}}, \mathbf{m}, \tau^M). \quad (25)$$

The slope p_I^h is given by Eq. (45) in *Iversen et al. (2012)*

$$p_I^h = \frac{\partial T^D}{\partial h_I}(\mathbf{h}, \hat{\mathbf{a}}, \mathbf{m}, \tau^M) + \psi_I^h \frac{\partial T^D}{\partial \tau}(\mathbf{h}, \hat{\mathbf{a}}, \mathbf{m}, \tau^M). \quad (26)$$

3.1.1. Half-offset vector with only one degree of freedom

The coverage of azimuthal directions for the half-offset vector \mathbf{h} may be too small to permit extraction of two slope components ψ_I^h to be input to kinematic time demigration. For this situation the (two-component) half-offset vector has effectively only one degree of freedom, so that we can parametrize the components of \mathbf{h} by a single scalar, σ . The quantity σ will then be a distance variable for traces belonging to a common-image gather in the migration-time domain or a common-midpoint gather in the recording-time domain. The parametrization reads

$$h_I(\sigma) = h(\sigma) n_I(\sigma), \quad (27)$$

where h is the magnitude of \mathbf{h} and $\mathbf{n} = (n_I)$ is a unit vector. The quantity σ is a distance variable for traces belonging to a common-image gather in the migration-time domain or a common-midpoint gather in the recording-time domain. The first derivative of h_I with respect to σ is

$$\frac{dh_I}{d\sigma} = \frac{dh}{d\sigma} n_I + h \frac{dn_I}{d\sigma}. \quad (28)$$

The slope mapping in Eq. (26) can now be restated

$$p^\sigma = \frac{\partial T^D}{\partial h_I} \frac{dh_I}{d\sigma} + \psi^\sigma \frac{\partial T^D}{\partial \tau}, \quad (29)$$

where we have introduced new, effective, slopes with respect to offset in the recording-time and migration-time domains

$$p^\sigma = \frac{\partial T}{\partial \sigma}, \quad \psi^\sigma = \frac{\partial \tau}{\partial \sigma}. \quad (30)$$

3.2. Algorithm for kinematic time migration

In the case of kinematic time demigration we first solved a system of two equations in two unknowns, i.e., the components of the aperture vector, followed by a straightforward evaluation of the diffraction-time function to obtain the output time. In the kinematic time migration situation there will still be three unknowns of the same type, the aperture vector components plus an output time, but now we will in general need to solve simultaneously a system of three equations in the three unknowns. The equations under consideration are given as Eqs (68) and (69) in *Iversen et al. (2012)*; we restate them here as

$$\frac{\partial T}{\partial x_I}(\mathbf{h}, \mathbf{x}) = \frac{\partial T^D}{\partial a_I}(\mathbf{h}, \hat{\mathbf{a}}, \mathbf{x} - \hat{\mathbf{a}}, \hat{\tau}), \quad (31)$$

$$T(\mathbf{h}, \mathbf{x}) = T^D(\mathbf{h}, \hat{\mathbf{a}}, \mathbf{x} - \hat{\mathbf{a}}, \hat{\tau}). \quad (32)$$

To describe the unknowns it is convenient to introduce a three-component vector (ζ) such that $\zeta_I = \hat{a}_I$ and $\zeta_3 = \hat{\tau}$.

We use Eqs (31) and (32) to formulate a three component vector function $g_i(\zeta)$ as follows

$$g_I(\zeta) = \frac{\partial T^D}{\partial a_I} - \frac{\partial T}{\partial x_I}, \quad (33)$$

$$g_3(\zeta) = T^D - T, \quad (34)$$

where function arguments on the right-hand side have been skipped. The lowercase index i and the uppercase index I take the respective values 1, 2, 3 and 1, 2. Equations (31) and (32) can then be restated

$$g_i(\zeta) = 0. \quad (35)$$

To solve the system of equations (35) we may proceed in the same way as described for Eq. (20). That means, to use a Newton-Raphson approach using derivatives $\partial g_i / \partial \zeta_j$. These derivatives are given explicitly as

$$\frac{\partial g_I}{\partial \zeta_J} = \frac{\partial^2 T^D}{\partial a_I \partial a_J} - \frac{\partial^2 T^D}{\partial a_I \partial m_J}, \quad (36)$$

$$\frac{\partial g_I}{\partial \zeta_3} = \frac{\partial^2 T^D}{\partial a_I \partial \tau}, \quad (37)$$

$$\frac{\partial g_3}{\partial \zeta_J} = \frac{\partial T^D}{\partial a_J} - \frac{\partial T^D}{\partial m_J}, \quad (38)$$

$$\frac{\partial g_3}{\partial \zeta_3} = \frac{\partial T^D}{\partial \tau}. \quad (39)$$

To obtain them, we have used that the time T and the slopes $\partial T / \partial x_I$ in Eqs (33) and (34) are invariant in kinematic time migration.

Knowing the output point $(\mathbf{h}, \hat{\mathbf{m}}, \tau)$ resulting from a kinematic time migration, it means that any requested partial derivative of the diffraction-time function T^D in the variables h_I , a_I , m_I , and τ_I may be computed. The natural next task is to obtain the slopes ψ_I^m and ψ_I^h at the point $(\mathbf{h}, \hat{\mathbf{m}}, \tau)$. For that, we reuse here Eq. (70) from *Iversen et al. (2012)*

$$\psi_I^h = \left(\frac{\partial T^D}{\partial \tau} \right)^{-1} \left(p_I^h - \frac{\partial T^D}{\partial h_I} \right), \quad (40)$$

$$\psi_I^m = \left(\frac{\partial T^D}{\partial \tau} \right)^{-1} \left(p_I^x - \frac{\partial T^D}{\partial m_I} \right). \quad (41)$$

For a half-offset vector with only one degree of freedom, see Eq. (27), Eq. (40) can be restated

$$\psi^\sigma = \left(\frac{\partial T^D}{\partial \tau} \right)^{-1} \left(p^\sigma - \frac{\partial T^D}{\partial h_I} \frac{dh_I}{d\sigma} \right). \quad (42)$$

4. SENSITIVITY OF KINEMATIC TIME MIGRATION WITH RESPECT TO THE TIME-MIGRATION VELOCITY MODEL

Considering one selected parameter ν of the time-migration velocity model, we study how the kinematic time migration is affected by a perturbation in ν . We investigate the first-order changes of the reflection location with respect to the time-migration velocity model (i.e. the derivatives $d\hat{m}_I/d\nu$ and $d\hat{\tau}/d\nu$), as well as the first-order changes of the reflection slopes with respect to the time-migration velocity model (i.e. the derivatives $d\psi_I^h/d\nu$).

In view of the model representation described in Appendix E, we take ν specifically as the node value

$$\nu = M_\lambda^{r,s,t}. \quad (43)$$

4.1. First-order changes of the reflection location with respect to the time-migration velocity model

The location of a reflection in the migration-time domain can be expressed in terms of the parameter ν as $(\mathbf{h}, \hat{\mathbf{m}}(\mathbf{h}, \nu), \hat{\tau}(\mathbf{h}, \nu))$. As shown in Fig. 3, we investigate how this location $(\hat{\mathbf{m}}, \tau)$ is affected, to first order, by a perturbation in ν .

Equation (35) yields a mapping from a location in the recording-time domain to a location in the migration-time domain. The relation is valid regardless of which time-migration velocity model we use. As a consequence, we have

$$\frac{dg_i}{d\nu} = 0, \quad (44)$$

for $i = 1, 2, 3$.

Also, we note that the time T and the slopes $\partial T/\partial x_I$ in Eqs (33) and (34) are insensitive to the value of ν . A differentiation of these equations with respect to ν therefore yields

$$\frac{d}{d\nu} \left(\frac{\partial T^D}{\partial a_I} \right) = 0, \quad (45)$$

$$\frac{dT^D}{d\nu} = 0. \quad (46)$$

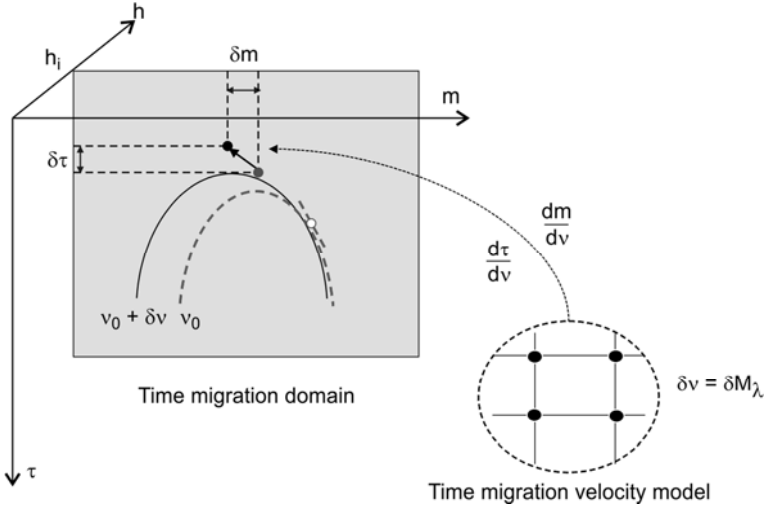


Fig. 3. Schematic overview (2D) of the first-order changes of the reflection location with respect to the time-migration velocity model. The symbol h_i signifies a specific common-offset section. In the figure, perturbation of a time-migration velocity model parameter (δv) leads to the position change (δm , $\delta \tau$) of the kinematic migration result: the kinematic migrated output is shifted from the grey point to the black point. These perturbations are described by the derivatives: dm/dv and $d\tau/dv$. The white point and the dashed straight line represent the input event point and its local slope. The grey point and the dashed grey curve represent the migrated point and its diffraction traveltimes curve by using the initial time-migration velocity of v_0 . Correspondingly, the black point and the solid black line represent the migrated point and its diffraction traveltimes curve by using the updated time-migration velocity $v_0 + \delta v$.

Equations (45) and (46) contain information to estimate the derivatives $d\hat{m}_I/dv$ and $d\hat{\tau}/dv$. The total derivative operator in these equations can be stated as

$$\frac{d}{dv} = \frac{\partial}{\partial v} + \frac{d\hat{a}_J}{dv} \frac{\partial}{\partial a_J} + \frac{d\hat{m}_J}{dv} \frac{\partial}{\partial m_J} + \frac{d\hat{\tau}}{dv} \frac{\partial}{\partial \tau}.$$

Using that the location (x_J) is insensitive to the value of v , the operator appears in the form

$$\frac{d}{dv} = \frac{\partial}{\partial v} + \frac{d\hat{m}_J}{dv} \left(\frac{\partial}{\partial m_J} - \frac{\partial}{\partial a_J} \right) + \frac{d\hat{\tau}}{dv} \frac{\partial}{\partial \tau}. \quad (47)$$

The specific partial derivatives of the diffraction time with respect to v are given in Appendix B.

Applying Eq. (47) in Eqs (45) and (46) yields

$$\left[\frac{d\hat{m}_J}{d\nu} \left(\frac{\partial}{\partial m_J} - \frac{\partial}{\partial a_J} \right) + \frac{d\hat{\tau}}{d\nu} \frac{\partial}{\partial \tau} \right] \left(\frac{\partial T^D}{\partial a_I} \right) = - \frac{\partial^2 T^D}{\partial \nu \partial a_I},$$

$$\left[\frac{d\hat{m}_J}{d\nu} \left(\frac{\partial}{\partial m_J} - \frac{\partial}{\partial a_J} \right) + \frac{d\hat{\tau}}{d\nu} \frac{\partial}{\partial \tau} \right] (T^D) = - \frac{\partial T^D}{\partial \nu},$$

and hence

$$\left(\frac{\partial^2 T^D}{\partial a_I \partial m_J} - \frac{\partial^2 T^D}{\partial a_I \partial a_J} \right) \frac{d\hat{m}_J}{d\nu} + \frac{\partial^2 T^D}{\partial a_I \partial \tau} \frac{d\hat{\tau}}{d\nu} = - \frac{\partial^2 T^D}{\partial \nu \partial a_I}, \quad (48)$$

$$\left(\frac{\partial T^D}{\partial m_J} - \frac{\partial T^D}{\partial a_J} \right) \frac{d\hat{m}_J}{d\nu} + \frac{\partial T^D}{\partial \tau} \frac{d\hat{\tau}}{d\nu} = - \frac{\partial T^D}{\partial \nu}. \quad (49)$$

Equations (48) and (49) can be restated as the matrix equation

$$\mathbf{A}\boldsymbol{\eta} = \mathbf{b}, \quad (50)$$

which includes the 3×1 matrix of unknowns

$$\boldsymbol{\eta} = \begin{bmatrix} \{d\hat{m}_J/d\nu\} \\ d\hat{\tau}/d\nu \end{bmatrix}, \quad (51)$$

and the 3×3 and 3×1 coefficient matrices

$$\mathbf{A} = \begin{bmatrix} \left\{ \frac{\partial^2 T^D}{\partial a_I \partial m_J} - \frac{\partial^2 T^D}{\partial a_I \partial a_J} \right\} & \left\{ \frac{\partial^2 T^D}{\partial a_I \partial \tau} \right\} \\ \left\{ \frac{\partial T^D}{\partial m_J} - \frac{\partial T^D}{\partial a_J} \right\} & \frac{\partial T^D}{\partial \tau} \end{bmatrix}, \quad (52)$$

$$\mathbf{b} = - \begin{bmatrix} \left\{ \partial^2 T^D / \partial \nu \partial a_I \right\} \\ \partial T^D / \partial \nu \end{bmatrix}. \quad (53)$$

In Eqs (51)–(53) brackets $\{ \}$ are used to signify the formation of a submatrix from an indexed component. The indices I and J in these equations take the values 1, 2.

Inversion of Eq. (50)

$$\boldsymbol{\eta} = \mathbf{A}^{-1} \mathbf{b} \quad (54)$$

will then yield the sought derivatives $d\hat{m}_I/d\nu$ and $d\hat{\tau}/d\nu$.

4.2. First-order changes of reflection slopes with respect to the time-migration velocity model

Apart from the investigation of the reflection position changes of a kinematic time migration output, we also need to study the variation of reflection slopes with respect to the perturbation of time-migration velocity. As presented in the Fig. 4, the derivative $d\psi_I^h/dv$ describes the sensitivity of the variation of the reflection slopes with respect to the velocity perturbation. In the following, we derive expressions for the first-order derivatives of reflection slopes in offset, resulting from a kinematic time migration, taken with respect to parameters of the time-migration velocity model.

4.2.1. Half-offset vector with two effective degrees of freedom

The most general mapping of reflection slopes ψ_I^h is given by Eq. (40), for which the half-offset vector has two degrees of freedom. We differentiate this equation with respect to the parameter v , which yields

$$\frac{d}{dv} \left(\frac{\partial T^D}{\partial h_I} + \frac{\partial T^D}{\partial \tau} \psi_I^h \right) = 0. \quad (55)$$

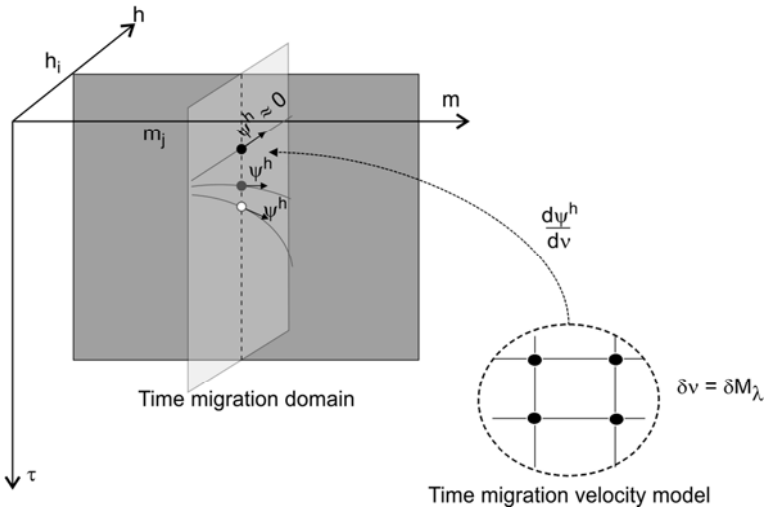


Fig. 4. Schematic overview (2D) of the first-order changes of the reflection (offset) slopes with respect to the time-migration velocity model. The symbols h_i and m_j signify, respectively, a common-offset section and a common-image gather. The figure shows, with a fixed input point from the recording domain, the perturbation of time-migration velocity model leads to the changes of the reflection slopes in the kinematic migration domain. The white and grey points represent the initial migrated point and the further migrated point after the velocity update, respectively. The black point resembles the ideal time-migration result ($\psi^h = 0$) derived from the optimal migration velocity.

Using the differentiation operator in Eq. (47), we obtain after some elaboration

$$\begin{aligned} \frac{d\psi_I^h}{d\nu} = & -\left(\frac{\partial T^D}{\partial \tau}\right)^{-1} \left\{ \frac{\partial^2 T^D}{\partial \nu \partial h_I} + \frac{\partial^2 T^D}{\partial \nu \partial \tau} \psi_I^h \right. \\ & + \frac{d\hat{m}_J}{d\nu} \left[\frac{\partial^2 T^D}{\partial m_J \partial h_I} - \frac{\partial^2 T^D}{\partial a_J \partial h_I} + \left(\frac{\partial^2 T^D}{\partial m_J \partial \tau} - \frac{\partial^2 T^D}{\partial a_J \partial \tau} \right) \psi_I^h \right] \\ & \left. + \frac{d\hat{\tau}}{d\nu} \left(\frac{\partial^2 T^D}{\partial \tau \partial h_I} + \frac{\partial^2 T^D}{\partial \tau^2} \psi_I^h \right) \right\}. \end{aligned} \quad (56)$$

4.3. Half-offset vector with one effective degree of freedom

When the half-offset vector has effectively only one degree of freedom, we take the mapping equation (42) as a starting point for obtaining a derivative of the slope along offset ψ^σ with respect the model parameter ν . The result is

$$\begin{aligned} \frac{d\psi^\sigma}{d\nu} = & -\left(\frac{\partial T^D}{\partial \tau}\right)^{-1} \left\{ \frac{\partial^2 T^D}{\partial \nu \partial h_I} \frac{dh_I}{d\sigma} + \frac{\partial^2 T^D}{\partial \nu \partial \tau} \psi^\sigma \right. \\ & + \frac{d\hat{m}_J}{d\nu} \left[\left(\frac{\partial^2 T^D}{\partial m_J \partial h_I} - \frac{\partial^2 T^D}{\partial a_J \partial h_I} \right) \frac{dh_I}{d\sigma} + \left(\frac{\partial^2 T^D}{\partial m_J \partial \tau} - \frac{\partial^2 T^D}{\partial a_J \partial \tau} \right) \psi^\sigma \right] \\ & \left. + \frac{d\hat{\tau}}{d\nu} \left(\frac{\partial^2 T^D}{\partial \tau \partial h_I} \frac{dh_I}{d\sigma} + \frac{\partial^2 T^D}{\partial \tau^2} \psi^\sigma \right) \right\}. \end{aligned} \quad (57)$$

In the calculation of Eq. (57), the partial derivatives related to the time-migration velocity model are needed. The detailed description of this calculation is described in Appendix F.

5. VELOCITY ESTIMATION FORMULATION

Once we formulated the kinematic time migration/demigration scheme and derived the Fréchet derivatives of kinematic time migration with respect to the time-migration velocity model, we are now able to set up an inversion scheme to estimate the time-migration velocity. In the following, we first review a common way to linearize a generally non-linear inversion problem, then we introduce the approach of iterative linearized inversion to estimate the optimal time-migration velocity model.

5.1. Linearization of the inversion problem

Consider a parameter vector \mathbf{v} containing N parameters belonging to a general data model (not necessarily related to seismics) and a vector function $\mathcal{D}(\mathbf{v})$ with M components

$$\mathcal{D}_m(\mathbf{v}) = d_m^{\text{true}} - d_m^{\text{pred}}(\mathbf{v}), \quad m = 1, \dots, M, \quad (58)$$

where d_m^{pred} is a quantity predicted on the basis of the model parameter vector \mathbf{v} , and d_m^{true} is the corresponding true (observed or idealized) quantity. Linearization of Eq. (58) with respect to a certain reference vector \mathbf{v}^0 yields

$$\mathcal{D}_m(\mathbf{v}) = \mathcal{D}_m(\mathbf{v}^0) + \frac{\partial \mathcal{D}_m}{\partial v_n}(\mathbf{v}^0)(v_n - v_n^0). \quad (59)$$

Furthermore, assume that $\mathcal{D}_m = 0$ for all values of the index m . The first-order update \mathbf{v} of the model parameter vector \mathbf{v}^0 must then satisfy the equation

$$\frac{\partial \mathcal{D}_m}{\partial v_n}(\mathbf{v}^0)(v_n - v_n^0) = -\mathcal{D}_m(\mathbf{v}^0). \quad (60)$$

Using the definition of the vector function \mathcal{D} in Eq. (58) we can alternatively write

$$\frac{\partial d_m^{\text{pred}}}{\partial v_n}(\mathbf{v}^0)(v_n - v_n^0) = d_m^{\text{true}} - d_m^{\text{pred}}(\mathbf{v}^0). \quad (61)$$

Equation (61) describes a system of M linear equations with N unknown parameters to be determined. The quantity $\partial d_m^{\text{pred}} / \partial v_n$ is commonly referred to as a Fréchet derivative.

5.2. Iterative linearized inversion

We use an iterative linearized inversion approach where the objective is to minimize the misfit between idealized and predicted slopes in the offset coordinates of the common-image gathers. The idealized slopes are zero and correspond to an optimal time-migration velocity model. It is remarked that such a model may not exist in practice, because of the limitations of time migration.

In our situation, the linear equations are obtained by adaption of Eq. (61), so that each slope event under consideration will give rise to two equations or a single equation, depending on the effective degree of freedom for the variation of the half-offset vector. The resulting equations for these two situations are, respectively,

$$\frac{\partial \psi_I^h}{\partial v_n}(\mathbf{v}^0)(v_n - v_n^0) = -\psi_I^h(\mathbf{v}^0), \quad I = 1, 2, \quad (62)$$

and

$$\frac{\partial \psi^\sigma}{\partial v_n}(\mathbf{v}^0)(v_n - v_n^0) = -\psi^\sigma(\mathbf{v}^0). \quad (63)$$

The Fréchet derivatives of our velocity estimation approach are partial derivatives of the reflection slope in the offset direction, ψ_I^h or ψ^σ , with respect to the parameters v_n , of the time-migration velocity model. These derivatives can be obtained using Eq. (56) or Eq. (57), respectively.

To avoid artificial jumps of the solution in areas with little data coverage, it is important to include regularization in the system of linear equations. We use Tikhonov regularization of zeroth, first, and second order (e.g., *Aster et al., 2013*). The first- and second-order regularization equations are included separately for each coordinate direction under consideration.

6. EXTRACTION OF LOCAL KINEMATIC PARAMETERS

With the above formulation, we are now able to estimate the time-migration velocity model based on the kinematic time migration and demigration by using the local kinematic parameters (traveltimes, slopes, and curvatures). In our method, considering the efficiency in implementation and the robustness in the calculation, we choose the gradient structure tensor (GST) method for the first-order kinematic parameters (slopes) extraction and quadratic gradient structure tensor (QST) method for the second-order kinematic parameters (curvatures) extraction. This operation is applied either on stacked data in the migration domain or the seismic stack in recording domains. In the following, we review the gradient structure tensor method and GST method for slope extraction. we also give the description of curvature extraction in Appendix D.

6.1. The gradient structure tensor

Structure tensors provide a description of the local structure in images using a tensor field. For a 3D image, this yields a 3×3 tensor matrix for each voxel. *Knutsson (1989)* showed that this tensor could be optimally obtained using six spatially oriented quadrature filters. The GST (*Bigun and Granlund, 1987*) is a simplified implementation of the structure tensor, where the tensor is estimated using three gradient filters.

For a 3D seismic cube, the GST is computed using gradient estimates along the three image dimensions ($\mathbf{x} = [x, y, t]^T$). The gradient tensor

$$\bar{\mathbf{T}} \equiv \overline{\mathbf{g}\mathbf{g}^T} \quad (64)$$

is the estimated covariance matrix of the gradient vector field

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} g_x(\mathbf{x}) \\ g_y(\mathbf{x}) \\ g_t(\mathbf{x}) \end{bmatrix}. \quad (65)$$

The gradients (g_x , g_y , g_t) along the three axes (x , y , t) are obtained by convolving the seismic data cube $I(\mathbf{x})$ with the derivative of the 3D Gaussian function $G(\mathbf{x}, \sigma)$:

$$g_i = I(\mathbf{x}) * \frac{\partial}{\partial x_i} G(\mathbf{x}, \sigma_g), \quad (66)$$

where σ_g is the gradient scale parameter. As the GST is sensitive to structures at different scales depending on σ_g , when applied to seismic data, we want the GST to be sensitive to reflections. This can be achieved by setting σ_g to match half the typical thickness of the reflections. (full width at half maximum of the Gaussian function). The GST is then computed as the smoothed outer product of the gradient vector

$$\bar{\mathbf{T}} = \begin{bmatrix} \overline{g_x^2} & \overline{g_y g_x} & \overline{g_t g_x} \\ \overline{g_x g_y} & \overline{g_y^2} & \overline{g_t g_y} \\ \overline{g_x g_t} & \overline{g_y g_t} & \overline{g_t^2} \end{bmatrix}, \quad (67)$$

where $\bar{\cdot}$ is the smoothing operator. The smoothed tensor elements are computed using the Gaussian window function

$$\bar{T}_{ij} = T_{ij} * G(\mathbf{x}, \sigma_T), \quad (68)$$

where σ_T is a parameter controlling the spatial smoothing. By having a large σ_T , the GST becomes more robust against noise. By having a smaller σ_T , more detailed information about the structure may be obtained.

6.2. Slope estimation

The local structure in the image can be analyzed by considering the eigenvalues ($\lambda_1 \geq \lambda_2 \geq \lambda_3$) and corresponding eigenvectors ($\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$) of $\bar{\mathbf{T}}$. Since $\bar{\mathbf{T}}$ is the estimated covariance matrix of the gradient vector field, the eigenvectors span the axes of the covariance ellipsoid. This means that \mathbf{v}_1 will have the same direction as the locally dominant direction of the gradient vector field. It is also possible to derive attributes from the eigenstructure (Randén *et al.*, 2000).

By considering the components of $\mathbf{v}_1 = [v_{1x}, v_{1y}, v_{1t}]$, the slopes (estimates of the derivatives) are given as

$$q_x = \frac{\widehat{\partial t}}{\partial x} = \frac{v_{1x}}{v_{1t}}, \quad (69)$$

$$q_y = \frac{\widehat{\partial t}}{\partial y} = \frac{v_{1y}}{v_{1t}}. \quad (70)$$

The eigenvector analysis is carried out for all locations in space, which produces the slope-fields.

7. NUMERICAL EXAMPLES

In this section, we demonstrate the viability of the proposed time-migration velocity estimation technique using three numerical examples. In the first example, we evaluate the derived formulation of kinematic time migration and the calculation of the Fréchet derivatives based on a constant time-migration velocity model. In the next, we present an example based on a 2D synthetic dataset. In the last example, we provide an application of time-migration velocity estimation on real 3D marine data from the Barents Sea.

7.1. Test of the Fréchet derivative formula for time-migration velocity estimation

First, we tested the kinematic migration using single event (point) mapping from the recording domain to the migration domain. As shown in Fig. 5, we used an input event point at CMP location $x = 2.5$ km, with travel time $T^D = 2.2676$ s and a half offset of $h = 1$ km. The corresponding event slopes were $p^x = 0.6840$ s km⁻¹ and $p^h = 0.0694$ s km⁻¹. In the example, a true time-migration velocity model with $S^M = 0.16$ s² km⁻² was known, and a time-migration velocity model with $S^M = 0.175$ s² km⁻² was used in the test. In Fig. 5, the diffraction traveltimes resulting from the true model and the test model are plotted, respectively, as solid and dashed black lines. During the test, we applied the kinematic migration to the selected input event in the recording domain based on the test and true models, and observed that the kinematic migration using the test migration velocity misplaced the reflection event in the migration domain and derived kinematic parameters, $m = 0.1889$ m, $\tau = 1.1011$ s, $\psi^m = 1.2692$ and $\psi^h = -0.0447$ s km⁻¹, while the kinematic migration using the true

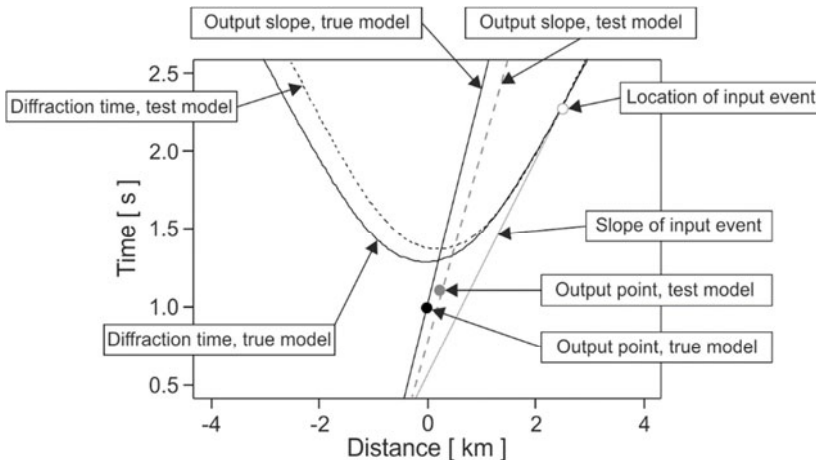


Fig. 5. Kinematic time-migration test using different velocity models. The white point represents an input event point in the recording domain. The black point represents the output event point of the kinematic migration based on the true velocity model, while the grey point represents the migrated output based on the test velocity model. The diffraction time functions using the true and test velocity models are represented by the solid and dashed black curves, respectively.

migration velocity correctly mapped the event back to the known diffraction point location, and derived the correct kinematic parameters, $m = 0$ km, $\tau = 1.0$ s, $\psi^m = 1.4009$ and $\psi^h = 0$ s km⁻¹ in the migration domain.

In the next example, we verified the accuracy of the Fréchet derivative approximations derived in Eqs (56) or (57). In the test, we first numerically generated the function $\psi^h(\nu)$ using the input migration velocity quantities $\nu = S^M$ and the reflection slope quantities in the offset direction ψ^h , derived from the kinematic migration (Fig. 6). Since the Fréchet derivatives of our constrained linear inversion are partial derivatives of the reflection slope in the offset direction, ψ_I^h or ψ^σ , with respect to the parameters ν_n of the time-migration velocity model, we could then calculate these derivatives by using Eqs (56) or (57). In Fig. 6, we plotted the linear approximation of $\psi^h(\nu)$ using an explicit formula for the derivative $\partial\psi^h/\partial\nu$ at velocity model coordinate $\nu = S^M = 0.175$ s² km⁻². The derived Fréchet derivative $\partial\psi^h/\partial\nu = -2.7389$ km s⁻¹, using the explicit formula in Eq. (56), is very close to the numerically-derived Fréchet derivative $\partial\psi^h/\partial\nu = -2.7390$ km s⁻¹. However, if we ignore the movement of the point (m, τ) in Eq. (56), the derived linear approximation of $\psi^h(\nu)$ at the selected velocity model coordinate gives the wrong slope approximation.

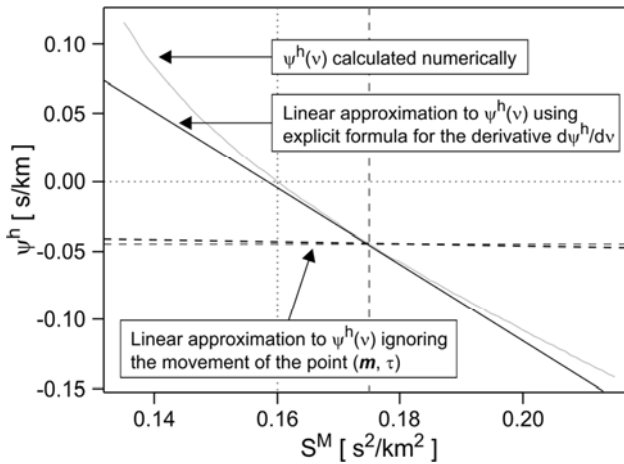


Fig. 6. Verification of the Fréchet derivative approximations. The grey curve represents the numerically-calculated function $\psi^h(\nu)$. The solid black line denotes the calculated Fréchet derivative based on Eq. (56) at the velocity model coordinate $\nu = S^M = 0.175$ s² km⁻² (vertical grey dashed line), while the dashed black line is the Fréchet derivative calculated for the same velocity-model coordinate, but ignoring the movement of the point (m, τ) .

7.2. 2D time-migration velocity estimation

In the 2D velocity estimation example, we used a 2D synthetic data set to demonstrate the proposed time-migration velocity estimation workflow (1), shown in Fig. 7. This is based on fully using all common-offset gathers, reflection surface picking and slope extraction in the migration domain, and iterations of the internal time-migration velocity estimation.

In this example, a 2D depth-velocity model (Fig. 8) was generated, consisting of five geological layers with predefined velocities (1500, 2000, 2500, 3000 and 3500 m s⁻¹), delineated by four interfaces. A surface with a syncline and anticline is defined on Interface 2, and a dipping surface is defined on Interface 3. The prestack shot-gather dataset was simulated by Kirchhoff modelling in NORSAR using a conventional 2D marine seismic geometry: shot interval 12.5 m, trace interval 25 m, minimal offset 100 m, maximum offset 3100 m, group number 121, and total number of shots 953.

Based on the simulated prestack dataset, we applied 2D Kirchhoff prestack time migration using an initial migration velocity field. To test the feasibility of the velocity estimation algorithm, we deliberately used a constant water velocity 1500 m s⁻¹ as an initial migration velocity model for the Kirchhoff prestack time migration. After the migration, the gradient structure tensor-based slopes extraction was applied to the migrated volume to derive slopes volumes, and a semi-automated seed-point-based horizon tracking was applied so as to derive the reflection traveltime surface in the common image point and offset domain. The corresponding slopes were extracted along the four interpreted surfaces in the migration domain.

Based on the derived kinematic parameters in the migration domain - the half-offset h , the common image point coordinate m , the migration time τ , and the reflection slopes ψ^m and ψ^h the kinematic demigration was applied to map those kinematic parameters to the recording domain and derive the half-offset h , the common midpoint coordinate x , the

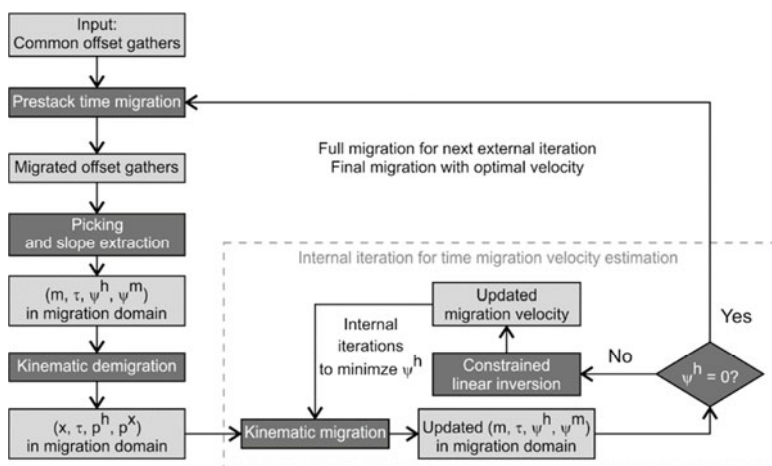


Fig. 7. Time-migration velocity estimation workflow 1.

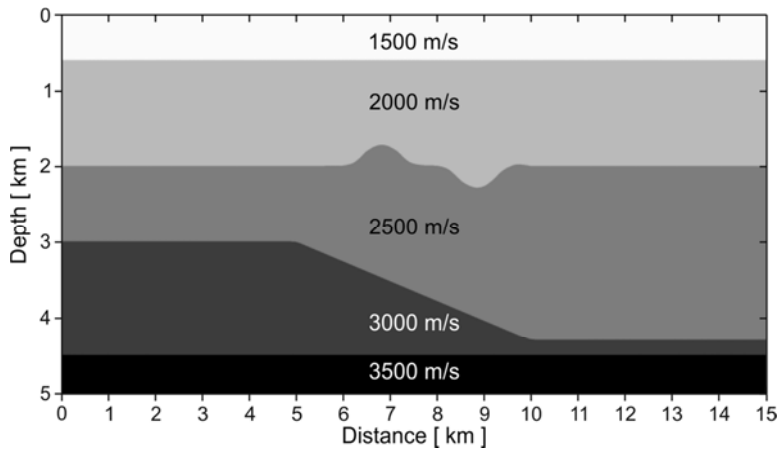


Fig. 8. 2D synthetic depth-velocity model.

reflection time T^D and the reflection slopes p^x and p^h . Based on the kinematic demigration, the reflection surfaces in the recording domain were recreated. Figure 9 shows the picked surfaces in the migration domain and the corresponding kinematic demigrated surfaces in the recording domain, selected from the finite-offset panels.

Using the input of kinematic parameters in the recording domain, the internal iterations for the time-migration velocity estimation were applied. Each iteration consisted of a kinematic migration with a defined velocity field, and the following constrained linear inversion for deriving the velocity update in order to minimise the reflection slope

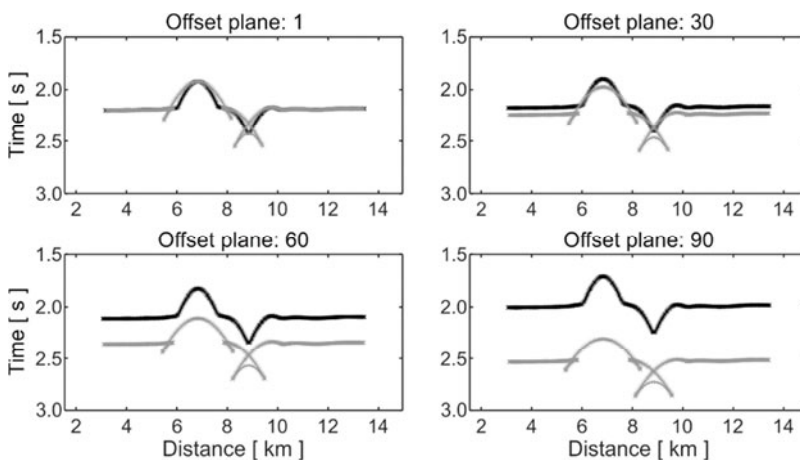


Fig. 9. Input finite surface planes in the migration domain (black), and kinematic demigrated finite surface planes in the recording domain (grey).

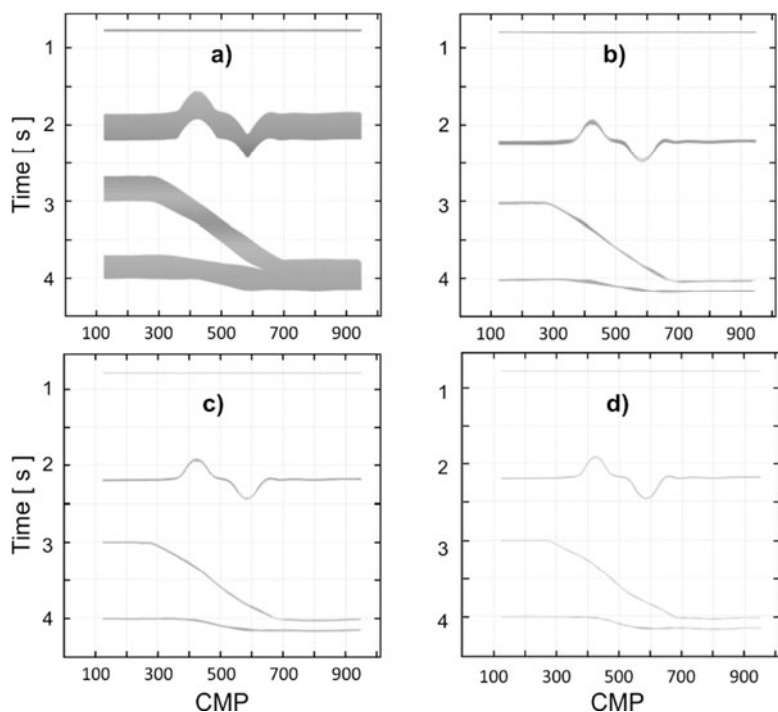


Fig. 10. Kinematically-migrated reflection surfaces based on the estimated time-migration velocity field. **a)** Reflection surface derived from initial water velocity; **b–d)** reflection surface derived from velocity-estimation iterations 1–3 (see Table 1 later), respectively. All surfaces in the finite-offset were projected onto the zero offset plane for this comparison. CMP – common midpoint.

ψ^h in the migration domain. In this synthetic example, we started the kinematic migration, based on the water velocity (1500 m s^{-1}), and the estimated time-migration velocity, by running three internal iterations. Figure 10 shows the kinematic migrated reflection surfaces based on the input/estimated time-migration velocity fields. We projected all the finite-offset surfaces onto the zero-offset plane to verify the improvement of the estimated time-migration velocity field. As mentioned earlier, the best estimated time-migration velocity field is able to flatten all the common image gathers in the offset direction, which is equivalent to zero for the quantity of reflection slope ψ^h in the migration domain. From the reduced residual moveout from Fig. 10, the rapid convergence of the algorithm can be seen.

The constrained linear inversion estimated time-migration velocity field was compared with the original depth-model-converted time-migration velocity field for the whole section, as shown in Fig. 11, and for the selected common-image-gather (CIG) locations in Fig. 12. The estimated time-migration velocity shows good consistency with the model converted reference time-migration velocity within the spatial coordinates 0 to 10 km. From 10 km to the end of the model, the estimated velocity presents a slightly slower

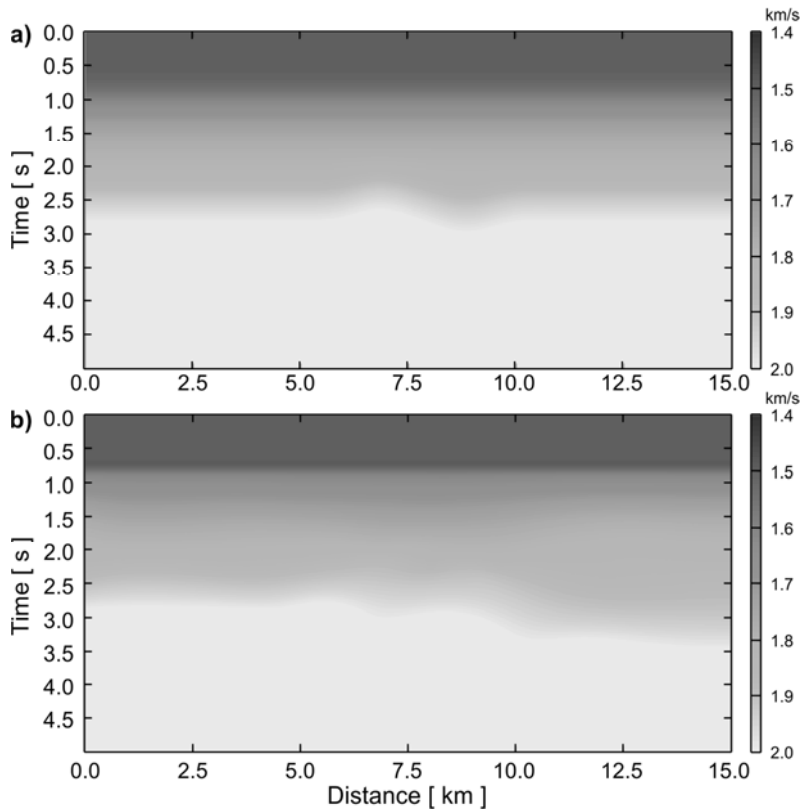


Fig. 11. **a)** Original time-migration velocity, and **b)** constrained linear inversion derived time-migration velocity.

velocity trend than the reference velocity, which might be the result of the sparse sampling between the second and third horizons at the right end of the original model.

In order to verify the quality of the derived time-migration velocity field, we applied Kirchhoff prestack time migration to the synthetic common-offset gathers using the derived time-migration velocity field. Figure 13 shows the selected CIG gathers after the migration. It can be seen that most of the events on the CIG gathers are flat following the migration using the estimated time-migration velocity. The final migrated stack, shown in Fig. 14, also proves the good quality of the estimated time-migration velocity field.

8. 3D TIME-MIGRATION VELOCITY ESTIMATION

In this final example, we employed a 3D marine dataset, acquired in the Barents Sea, to demonstrate the time-migration velocity estimation in 3D scenarios. The dataset contains the migrated stacked 3D volume, the interpreted geological horizons, the pre-migration CMP gathers, and the time-migration velocity field derived from the

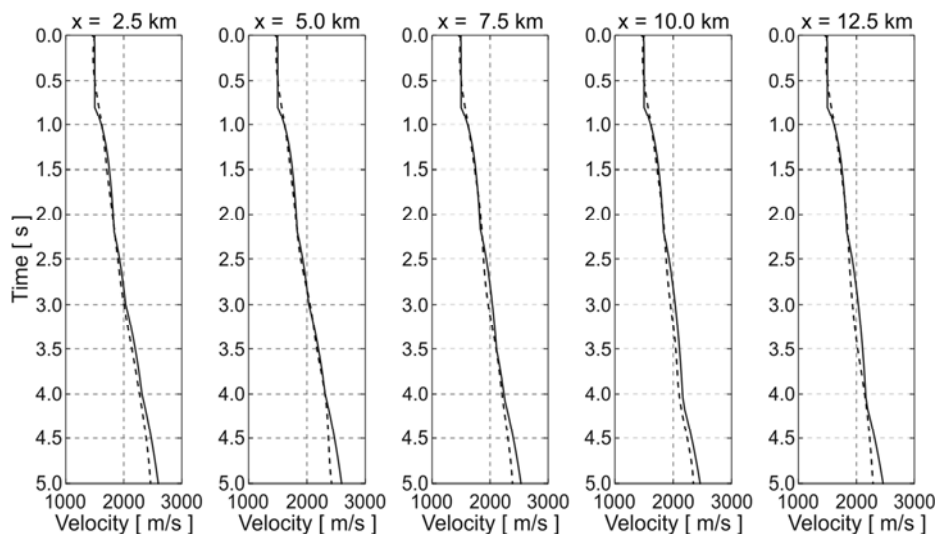


Fig. 12. Comparison of derived time-migrated velocity (dashed curve) and time-migration velocity estimated by method of Dix (1955) (solid curve), at selected common-image-gather locations.

conventional migration velocity estimation. As the migrated common image gathers were not available for the study, we modified the prior workflow (1) of the time-migration velocity estimation demonstrated in the 2D example, and proposed an alternative version of the workflow (2) (Fig. 15) for the 3D time-migration velocity estimation. Instead of the application of an initial prestack time migration to the input data, and the subsequent picking of each migrated offset volume, the algorithm uses the interpreted key reflectors from the time-migrated zero-offset volume as the input and replaces the reflection surface picking and slope extraction in the migration domain with the reflection surface modelling, fitting and slope extraction in the recording domain. In the following 3D numerical example, we used the provided migration velocity as the reference velocity model, and derived the demigrated zero-offset kinematic parameters. Using the subsequent application of reflection surface modeling, seismic to surface fitting and slope extraction, we estimated the 3D time-migration velocity based on the internal iteration of nonlinear constrained 3D velocity inversion. Ultimately, the derived migration velocity field was compared with the reference velocity model to demonstrate the effectiveness of the proposed 3D time-migration velocity estimation approach.

The 3D marine dataset was acquired using 12 streamers separated by 75 m and a dual-source configuration with a shot point interval of 18.75 m. The data on each cable was recorded using 564 receivers per streamer at interval of 12.5 m (Fig. 16a). The sorted 3D CMP dataset, a prestack time-migrated stack volume, the 3D time-migration velocity field and the interpreted horizons from the operating company were used in this numerical example. From the distribution of source-receiver offsets it was clear that we should use a scalar representation of the half-offset vectors (one degree of freedom). Thereby, for the 3D experiment the slope with respect to half offset is also a scalar quantity.

Based on legacy seismic processing, four horizons (Fig. 16b) were interpreted based on the migrated stack volume, which represents the geological interfaces of the seabed, top Triassic, intra-Triassic and top Permian respectively. The original surfaces covered a 300-km² area. In this example, we selected an area of interest of 25 km² ($x = 1.625\text{--}6.625$ km, $y = 0.375\text{--}5.375$ km) for the time-migration velocity estimation.

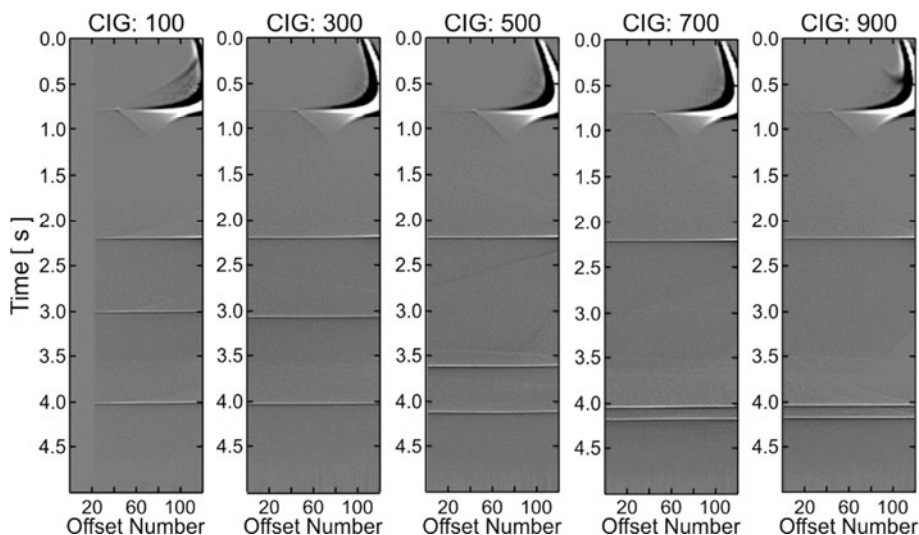


Fig. 13. Common-image-gathers (CIGs) migrated using derived migration velocity.

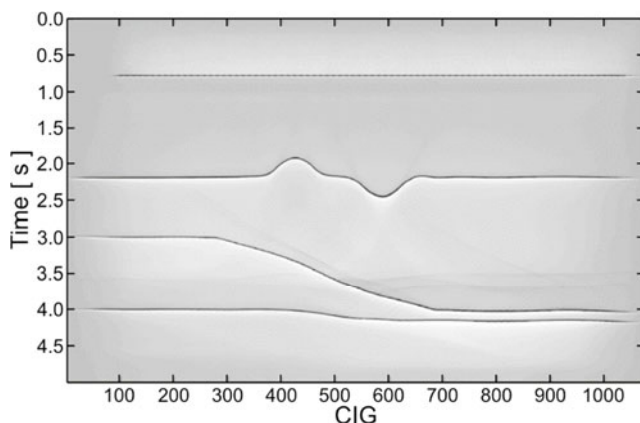


Fig. 14. Migration stacked image from prestack time-migration with derived time-migration velocity. CIG - common-image-gather.

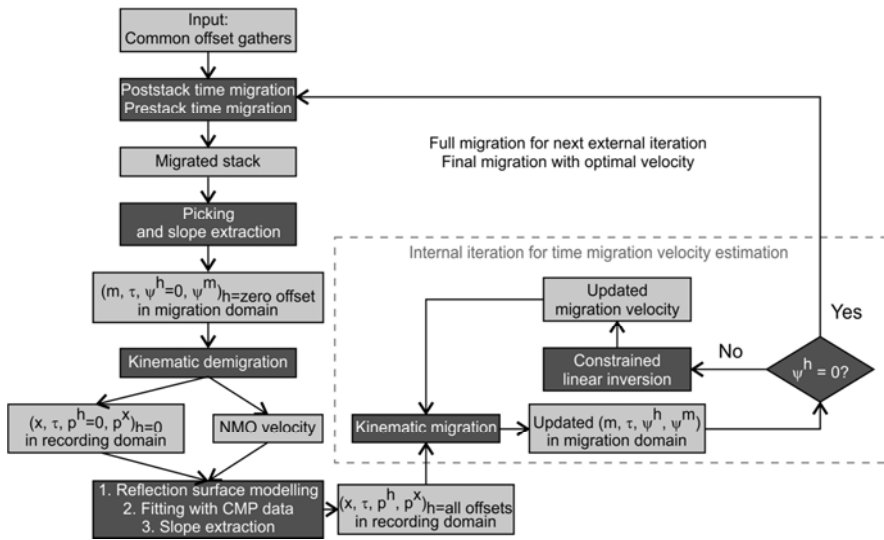


Fig. 15. Time-migration velocity estimation workflow 2.

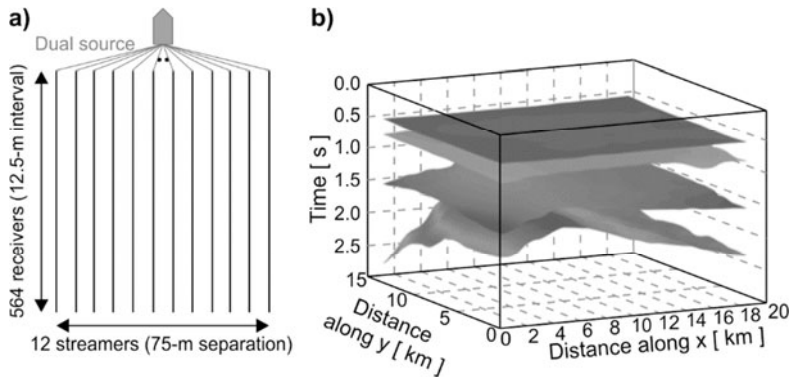


Fig. 16. **a)** 3D acquisition geometry in the survey; **b)** 3D surfaces picked from zero-offset migrated volume, which represent the geological interfaces of the seabed, top Triassic, intra-Triassic, and top Permian, from top to bottom respectively.

In the first step of the algorithm (Fig. 15), we proposed applying an initial prestack or poststack time migration using a raw time-migration velocity field, derived either from stacking velocity analysis or a manually-defined velocity trend, then applying the key reflections picking and slope extraction based on the migrated zero-offset dataset. In this example, based on the provided prestack time-migration (PSTM) volume and the interpreted key reflection horizons, we first derived the kinematic attribute slope ψ^m using the gradient structure tensor method and extracted the slopes along the interpreted surfaces to derive the zero-offset kinematic parameters $(m, \tau, \psi^h=0, \psi^m)$ in the migration domain. Kinematic demigration was applied subsequently to generate the zero-

offset kinematic parameters in the recording domain (\mathbf{x} , T^D , $p^h = 0$, \mathbf{p}^x) by employing the original prestack time-migration velocity. In Fig. 17, the fourth zero-offset surface in the migration domain, the demigrated surface in the recording domain and the corresponding slope ψ^m are displayed as an example.

Next, in order to obtain the kinematic parameters of the finite-offsets in the recording domain, we introduced a fast-track approach to reflection-surface modelling and surface-to-seismic fitting in the recording domain. In this process, referring to the derivation of Iversen *et al.* (2012) (Eq. (84)), we derived the normal moveout matrices \mathbf{S}^{NMO} based on the migration velocity S^M and zero-offset slope \mathbf{p}^x . We then modelled the traveltimes surfaces of the finite-offset reflections (Fig. 18) using the CMP approximation. The NMO matrices \mathbf{S}^{NMO} , zero-offset reflection surfaces T^0 , and offset vectors \mathbf{h} extracted from the CMP gathers were used in this calculation.

The generated finite-offset reflection surfaces in the recording domain are shown in Fig. 18. Meanwhile, considering that the CMP traveltime equation is under the assumption that the reflection travel time, as the function of the offset, follows hyperbolic trajectories, which are not valid in the case of complex overburden structures that give rise to strong lateral variations, in this case, we assumed the gentle velocity variation within the migration aperture and only tackled the time-migration velocity estimation for the isotropic medium. Thus, at next step in traveltimes surfaces/curves to seismic fitting is needed to match the derived reflection curves to the seismic events on the CMP gathers. A semblance-based parameter searching approach was applied in this fitting process. The fitting was applied to the selected CMP locations using the defined grid size of 500×500 m. Once the traveltime was calculated at the selected offset and CMP locations, the individual finite-offset surfaces were created using the bi-cubic spline interpolation based on the local traveltime control points. The original modelled reflection events and the fitted reflection events were overlaid on the corresponding CMP gather, and are shown in Fig. 19.

To derive the finite-offset kinematic parameters in the recording domain, polynomial fitting and slope extraction along the selected directions were applied on each reflection surface. The second order polynomial and cubic smoothing spline were used to fit and derive the slopes \mathbf{p}^x and \mathbf{p}^h , respectively. Figure 20 shows the reflection surfaces and their corresponding extracted slopes on the selected offset.

In the next step, based on the derived finite-offset kinematic parameters (\mathbf{h} , \mathbf{x} , T^D , \mathbf{p}^h , \mathbf{p}^x) in the recording domain, we applied the 3D kinematic migration and constrained the linear inversion in order to estimate the time-migration velocity. As mentioned in the section on the velocity estimation formulation, the internal time-migration velocity estimation is formulated as an iteration of the kinematic time migration and the constrained linear inversion. The kinematic time migration is applied on a defined velocity model to map the kinematic parameters from the recording domain to the migration domain, while the constrained linear inversion provides the time-migration velocity perturbation that minimises the misfit between the idealised slope $\mathbf{p}^h = 0$ and the current slope \mathbf{p}^h in the common-image gathers. In the 3D time-migration velocity estimation example, we selected a testing area ($5 \text{ km} \times 5 \text{ km} \times 2 \text{ s}$) for the velocity field estimation. Unlike using the constant-velocity model in the 2D time-migration velocity

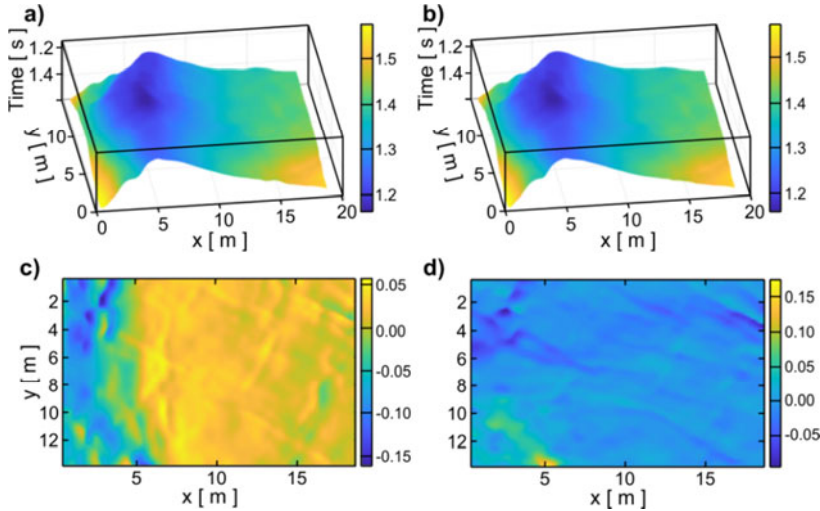


Fig. 17. Input and output of zero-offset kinematic demigration, and the derived slope vector p^x in the recording domain. **a)** Interpreted fourth surface in the migration domain; **b)** corresponding demigrated surface in the recording domain; **c)** x component of the slope vector p^x ; **d)** y component of the slope vector p^y .

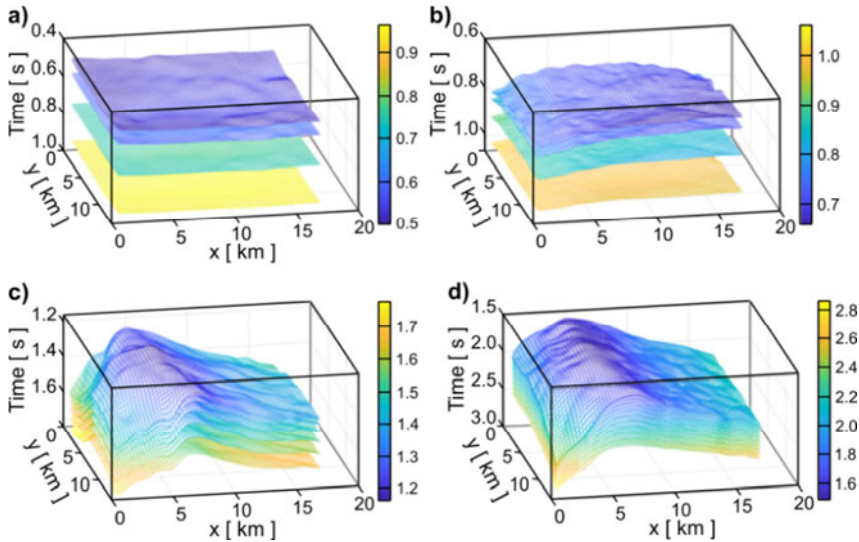


Fig. 18. Modelled finite-offset reflection surfaces in the recording domain. **a)** First finite-offset reflection surfaces using selected offset planes from the zero offset panel to the maximal offset panel of 15 with a step of 5 units in the panel index. **b)** Second finite-offset reflection surfaces using the same panel selection as under a). **c)** Third finite-offset reflection surfaces using selected offset planes from the zero offset panel to the maximal offset panel of 30 with step 5. **d)** Fourth finite-offset reflection surfaces with selected offset planes from the zero offset panel to the maximal offset panel of 60 with step 5.

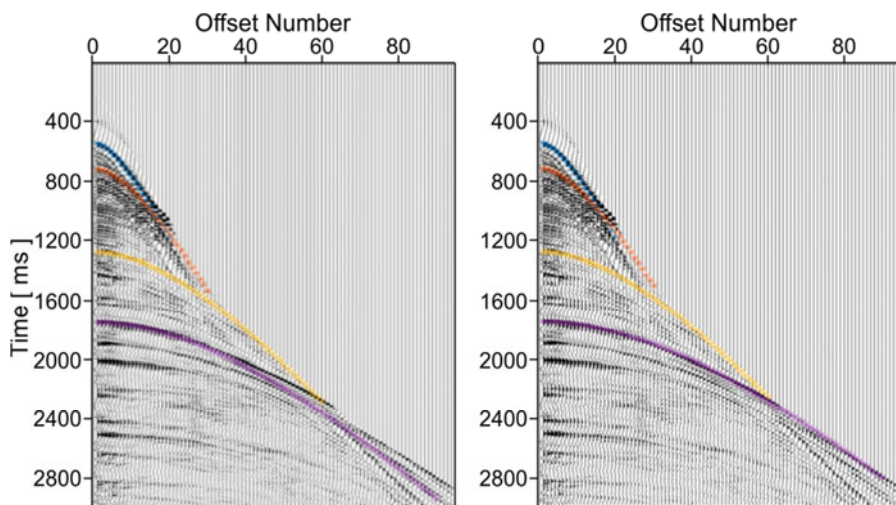


Fig. 19. Modelled reflection events (in colors) overlain on the common-midpoint (CMP) gather (left), and fitted reflection events overlain on the CMP gather (right).

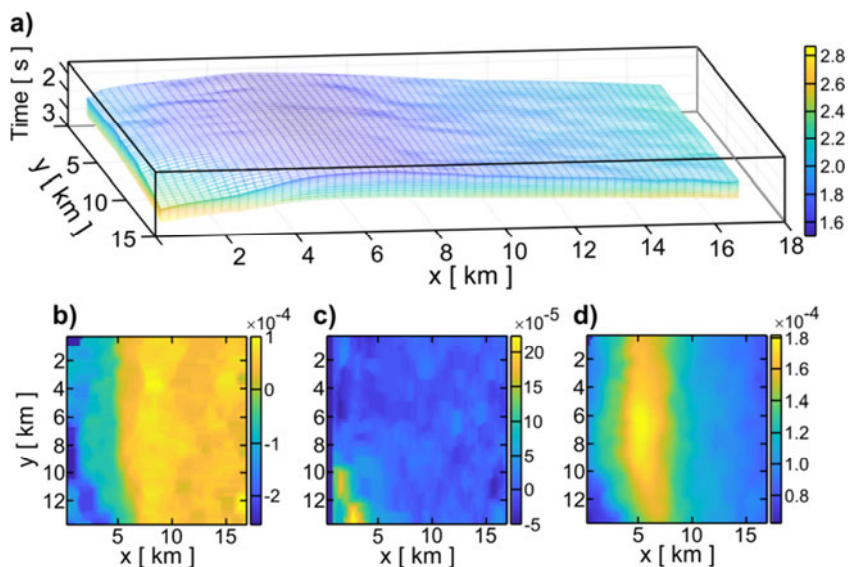


Fig. 20. Finite-offset reflection surfaces of the fourth surface from Fig. 16b and the corresponding extracted slopes on the tenth offset panel. **a)** Finite-offset reflection surfaces; **b)** the x -component of the slope vector p^x ; **c)** the y -component of the slope vector p^y ; **d)** the scalar slope p^h .

estimation example, we derived an initial 3D velocity model from a previous 2D time-migration velocity estimation using a specified sparse grid sampling ($1 \text{ km} \times 1 \text{ km} \times 0.1 \text{ s}$), which was used to constrain and stabilise the 3D linear inversion. The derived initial velocity field is shown in Fig. 21.

In the 3D time-migration velocity estimation, we applied three iterations of velocity inversion using the setting of variant grid sizes. The grid parameters of the migration velocity used for each iteration are listed in Table 1. The regularisation condition of the

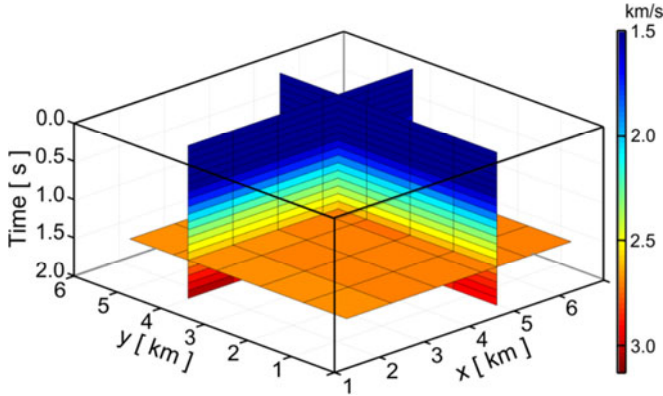


Fig. 21. Initial 3D velocity field model ($5 \text{ km} \times 5 \text{ km} \times 2 \text{ s}$).

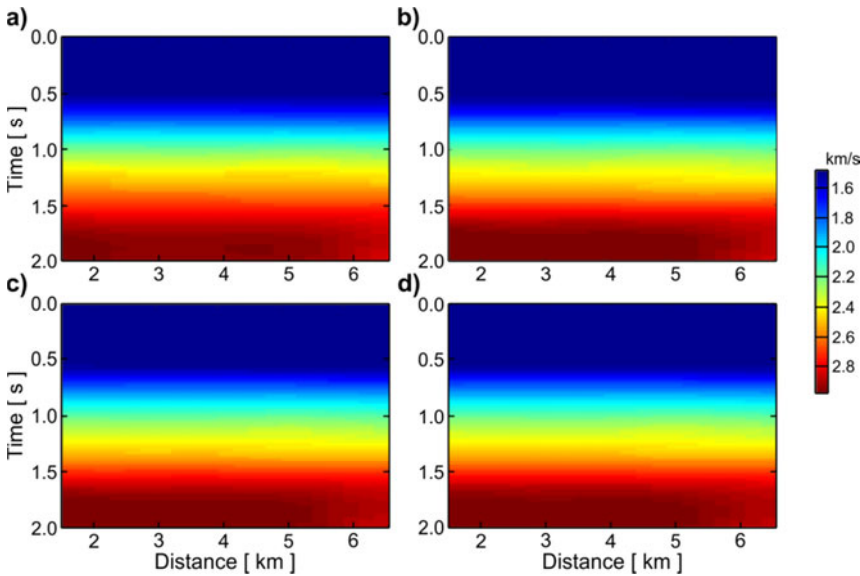


Fig. 22. Time-migration velocity from the 3D velocity estimation (selected inline for $x = 3.4 \text{ km}$): **a)** initial; **b)** from the first iteration, **c)** from the second iteration, and **d)** from the third iteration of the velocity estimation.

Table 1. Grid parameter setting for 3D time-migration velocity estimation.

	x Sampling [m]	y Sampling [m]	Time Sampling [ms]
Iteration 1	1000	1000	100
Iteration 2	500	500	50
Iteration 3	250	250	25

first order of Tikhonov using an adaptive damping scheme, was applied to all the iterations. The initial velocity model, and the derived migration velocities of each iteration, are shown in Fig. 22 for selected inline location ($x = 3.4$ km).

In the last step, for quality controlling the derived 3D time-migration velocity field, we used the reference time-migration velocity field, which was derived from conventional migration velocity analysis as a benchmark against which to compare the velocity field derived from our approach. The two velocity volumes are compared in Fig. 23. The selected inline and xline velocity profiles are compared in Figs 24 and 25, respectively. These two velocity fields were also compared at randomly-selected CIG locations, as shown in Fig. 26. From these figures, it can be clearly seen that the inversion-generated velocity field has good consistency with the velocity derived by the conventional approach, especially within the time range 0–1.6 s. In the deeper part, from 1.6 to 2 s, it can be observed that there is a mismatch between the inversion-derived velocity and the conventional-approach-derived velocity. The inversion-derived velocity gives a slower velocity than the conventional approach; however, upon investigation of the locations that were mismatched, we found that uplift of the bottom horizon and regularisation were the main reasons, and that, potentially, if more deep horizons were included, the version might better constrain the inversion, giving an even better velocity estimation at the bottom.

9. CONCLUSIONS

In this paper, we have presented a technique for time-migration velocity estimation, which is based on known reflection traveltimes and local slopes. We have generalised the time-migration velocity estimation using nonlinear mapping processes based on kinematic time migration and demigration, and derived, to a great detail, the involved Fréchet derivatives; that is, the partial derivatives of the local reflection slopes in the offset direction with respect to the time-migration velocity model. Ultimately, a system of linear equations was applied iteratively in order to solve the nonlinear velocity update.

Using the proposed scheme, we have generalised both the 2D and 3D homogeneous and heterogeneous seismic time-migration velocity estimation approaches based on reflection traveltimes and local slopes. The scheme is applied to both conventional narrow- and wide-azimuth acquisition geometries. The proposed time-migration velocity estimation approach is, in general, more accurate than conventional methods because it better honours the spatial variations of the diffraction time function. Through numerical examples, we have shown that the new approach is robust. The optimal migration velocity was obtained within the internal velocity estimation loop, and was subsequently used for final PSTM. Regarding reflection traveltimes picking, we proposed extracting the local

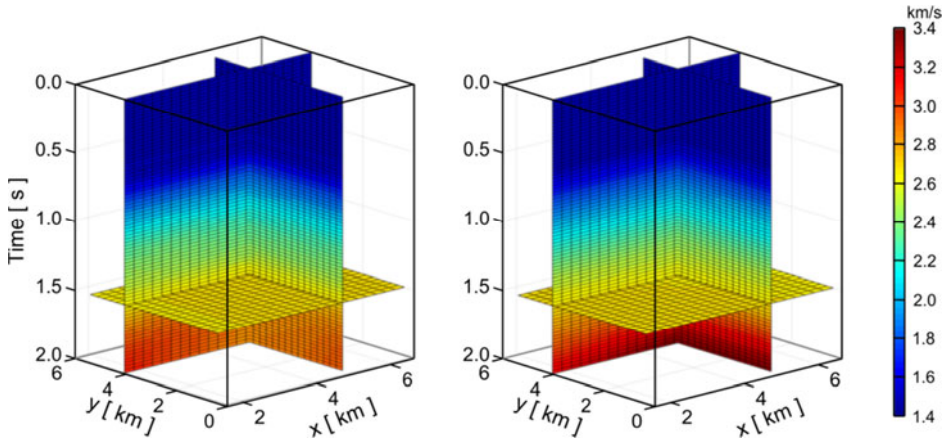


Fig. 23. Comparison of 3D time-migration velocity fields derived from the kinematic time migration and demigration approach (left), and the conventional migration velocity analysis approach (right).

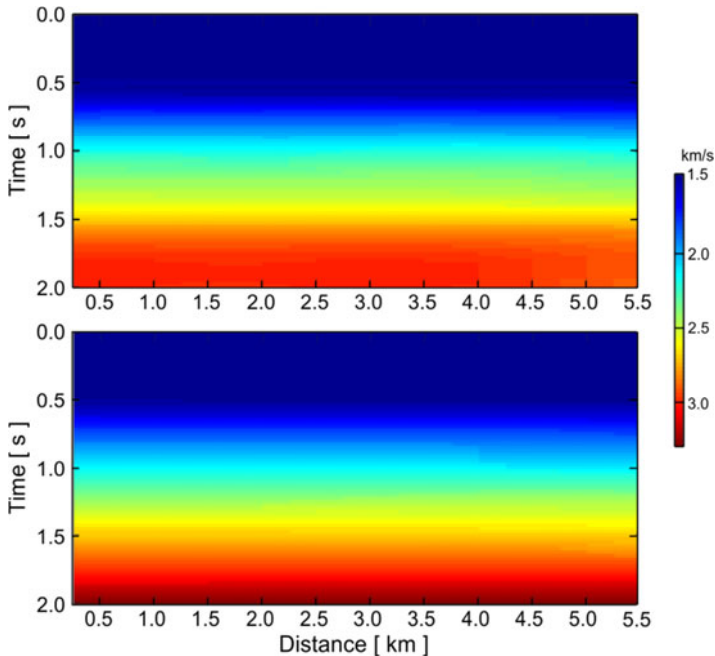


Fig. 24. Comparison of the velocity-estimation derived time-migration velocity field (top) and the conventional migration-velocity-analysis-derived time-migration velocity field (bottom). The plot corresponds to the inline $x = 3$ km, see Fig. 21.

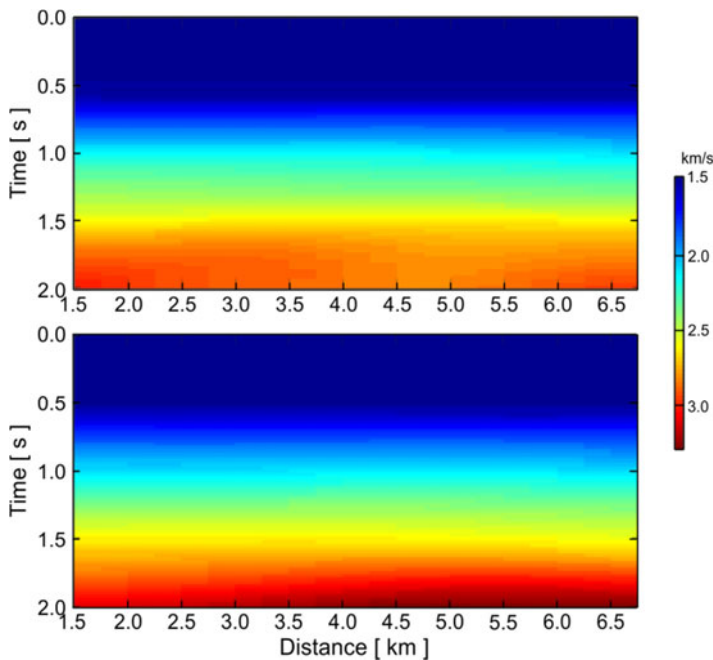


Fig. 25. The same as in Fig. 24, but for $y = 1.65$ km.

reflection slopes using the technique of gradient structure tensor and semi-automatic reflection horizon picking in the prestack migration domain in order to utilise the migrated prestack data with a higher resolution and signal-to-noise ratio.

In the numerical experiments, the theoretical formulations were tested using three examples. In the first example, we verified the Fréchet derivatives used for the time-migration velocity estimation. The presented examples of the comparison between the analytical and numerical derivatives of both the kinematic migration and Fréchet derivatives proved the accuracy of the theoretical derivation. In the next example, we demonstrated the time-migration velocity estimation on a 2D dataset. The experimental dataset was initially generated by Kirchhoff modelling. In the subsequent velocity estimation, the generated synthetic datasets were migrated by PSTM using an initial migration velocity model, and then the reflection traveltime picks and corresponding local slopes were derived. The following applied internal velocity updates showed a rapid convergence of the linear velocity iterations. The accuracy of the derived 2D time-migration velocity was also proved by the flatness of the final PSTM-migrated gather and the reasonable quality of the PSTM-migrated stack image. In the last numerical example, we demonstrated the feasibility of applying the derived scheme for the estimation of time-migration velocity to a 3D marine field dataset. In this example, as we were short of a PSTM-migrated prestack dataset, only obtaining the migrated stacked dataset, the corresponding integrated reflection horizons, and legacy PSTM migration-velocity field, we used a workaround scheme to derive the invariant reflection traveltimes and local

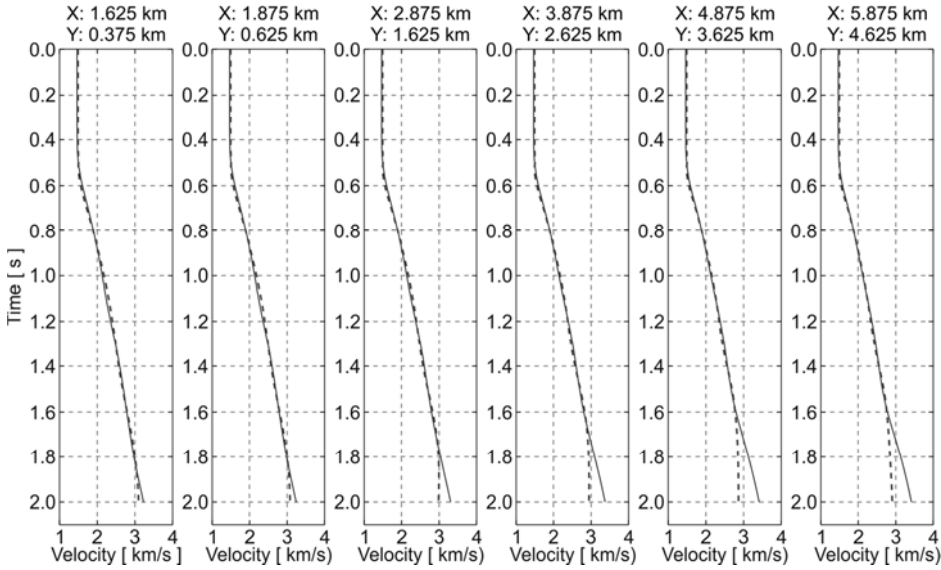


Fig. 26. Comparison of velocity-estimation derived time-migration velocity field (dashed curve) and the conventional migration-velocity-analysis-derived time-migration velocity field (solid curve) for randomly selected locations.

slopes in the recording domain, and applied the velocity estimation using the input from selected finite-offset planes. Although we understand that the picking in the recording domain may have included artifacts from the complex geological region, the experiment-derived migration velocity volume still showed good consistency with the velocity derived from conventional migration-velocity analysis. This proved the feasibility of the 3D time-migration velocity estimation formulation.

In general, considering the initial depth-velocity model is still crucial for most of the depth-domain tomography algorithms, we believe that the proposed technique of time-migration velocity estimation, using nonlinear kinematic migration/demigration solvers based on Fréchet derivatives, will improve the accuracy of velocity model estimation and the resulting final image in both the time and depth domains.

APPENDIX A

FIRST- AND SECOND-ORDER DERIVATIVES OF DIFFRACTION TIME WITH RESPECT TO MIGRATION-TIME COORDINATES AND APERTURE

The diffraction-time function relies on migration-time coordinates and aperture - totally seven independent variables. We specify the first- and second-order derivatives in these variables.

To simplify the formulations, define the coordinate vector $\xi = (\xi_i)$ such that $\xi_1 = m_1$, $\xi_2 = m_2$, $\xi_3 = \tau$, and the coordinate vector $\gamma = (\gamma_r)$, with $\gamma_1 = h_1$, $\gamma_2 = h_2$, $\gamma_3 = a_1$, $\gamma_4 = a_2$.

We observe that the first-order relations between source- and receiver-offset components (h_I^S and h_I^R) and the components γ_r (h_J and a_J) are given by

$$\frac{\partial h_I^S}{\partial h_J} = -\delta_{IJ}, \quad \frac{\partial h_I^S}{\partial a_J} = -\delta_{IJ}, \quad (\text{A.1})$$

and

$$\frac{\partial h_I^R}{\partial h_J} = -\delta_{IJ}, \quad \frac{\partial h_I^R}{\partial a_J} = -\delta_{IJ}. \quad (\text{A.2})$$

Once we know the derivative of T^S and T^R with respect to ξ_i and/or γ_r , the corresponding derivative of T^D results by simple summation.

A.1. Derivatives of one-way time

First, consider the one-way time function T^S . Straightforward differentiation of the first formula in Eq. (7) yields the first-order derivatives

$$\frac{\partial T^S}{\partial \xi_m} = \frac{1}{2T^S} \left[\frac{\xi_3}{2} \delta_{3m} + \frac{\partial}{\partial \xi_m} \left(S_{IJ}^{M(2)} \right) h_I^S h_J^S + \frac{\partial}{\partial \xi_m} \left(S_{IJKL}^{M(4)} \right) h_I^S h_J^S h_K^S h_L^S \right], \quad (\text{A.3})$$

$$\frac{\partial T^S}{\partial \gamma_r} = \frac{\partial T^S}{\partial h_M^S} \frac{\partial h_M^S}{\partial \gamma_r}, \quad \frac{\partial T^S}{\partial h_M^S} = \frac{1}{T^S} \left(S_{IM}^{M(2)} h_I^S + 2S_{IJKM}^{M(4)} h_I^S h_J^S h_K^S \right), \quad (\text{A.4})$$

with $\partial h_M^S / \partial \gamma_r$ given by Eq. (A.1).

For the second-order derivatives we find

$$\begin{aligned} \frac{\partial^2 T^S}{\partial \xi_m \partial \xi_n} = \frac{1}{2T^S} \left[\frac{\partial^2}{\partial \xi_m \partial \xi_n} \left(S_{IJ}^{M(2)} \right) h_I^S h_J^S + \frac{\partial^2}{\partial \xi_m \partial \xi_n} \left(S_{IJKL}^{M(4)} \right) h_I^S h_J^S h_K^S h_L^S \right] \\ - \frac{1}{T^S} \frac{\partial T^S}{\partial \xi_m} \frac{\partial T^S}{\partial \xi_n}, \end{aligned} \quad (\text{A.5})$$

$$\frac{\partial^2 T^S}{\partial \gamma_r \partial \gamma_s} = \frac{1}{T^S} \left(S_{MN}^{M(2)} + 6S_{IJMN}^{M(4)} h_I^S h_J^S \right) \frac{\partial h_M^S}{\partial \gamma_r} \frac{\partial h_N^S}{\partial \gamma_s} - \frac{1}{T^S} \frac{\partial T^S}{\partial \gamma_r} \frac{\partial T^S}{\partial \gamma_s}, \quad (\text{A.6})$$

$$\begin{aligned} \frac{\partial^2 T^S}{\partial \xi_m \partial \gamma_r} = \frac{1}{T^S} \left[\frac{\partial}{\partial \xi_m} \left(S_{IN}^{M(2)} \right) h_I^S + 2 \frac{\partial}{\partial \xi_m} \left(S_{IJKN}^{M(4)} \right) h_I^S h_J^S h_K^S \right] \frac{\partial h_N^S}{\partial \gamma_r} \\ - \frac{1}{T^S} \frac{\partial T^S}{\partial \xi_m} \frac{\partial T^S}{\partial \gamma_r}. \end{aligned} \quad (\text{A.7})$$

To obtain formulas for the corresponding derivatives of the one-way time T^R , simply substitute the superscript R for S in Eqs (A.3)–(A.7).

A.2. Derivatives of one-way time for rotationally symmetric diffraction moveout

If the diffraction-time moveout is rotationally symmetric, Eqs (A.3)–(A.7) are recast as

$$\frac{\partial T^S}{\partial \xi_m} = \frac{1}{2T^S} \left[\frac{\xi_3}{2} \delta_{3m} + \frac{\partial S^{M(2)}}{\partial \xi_m} (h^S)^2 + \frac{\partial S^{M(4)}}{\partial \xi_m} (h^S)^4 \right], \quad (\text{A.8})$$

$$\frac{\partial T^S}{\partial \gamma_r} = \frac{\partial T^S}{\partial h_M^S} \frac{\partial h_M^S}{\partial \gamma_r}, \quad \frac{\partial T^S}{\partial h_M^S} = \frac{1}{T^S} \left[S^{M(2)} + 2S^{M(4)} (h^S)^2 \right] h_M^S, \quad (\text{A.9})$$

$$\frac{\partial^2 T^S}{\partial \xi_m \partial \xi_n} = \frac{1}{2T^S} \left[\frac{\partial^2 S^{M(2)}}{\partial \xi_m \partial \xi_n} (h^S)^2 + \frac{\partial^2 S^{M(4)}}{\partial \xi_m \partial \xi_n} (h^S)^4 \right] - \frac{1}{T^S} \frac{\partial T^S}{\partial \xi_m} \frac{\partial T^S}{\partial \xi_n}, \quad (\text{A.10})$$

$$\frac{\partial^2 T^S}{\partial \gamma_r \partial \gamma_s} = \frac{1}{T^S} \left[S^{M(2)} + 6S^{M(4)} (h^S)^2 \right] \frac{\partial h_M^S}{\partial \gamma_r} \frac{\partial h_M^S}{\partial \gamma_s} - \frac{1}{T^S} \frac{\partial T^S}{\partial \gamma_r} \frac{\partial T^S}{\partial \gamma_s}, \quad (\text{A.11})$$

$$\frac{\partial^2 T^S}{\partial \xi_m \partial \gamma_r} = \frac{1}{T^S} \left[\frac{\partial S^{M(2)}}{\partial \xi_m} + 2 \frac{\partial S^{M(4)}}{\partial \xi_m} (h^S)^2 \right] h_N^S \frac{\partial h_N^S}{\partial \gamma_r} - \frac{1}{T^S} \frac{\partial T^S}{\partial \xi_m} \frac{\partial T^S}{\partial \gamma_r}. \quad (\text{A.12})$$

APPENDIX B

FIRST-ORDER DERIVATIVES OF DIFFRACTION TIME WITH RESPECT TO PARAMETERS OF THE TIME-MIGRATION VELOCITY MODEL

We show how one can calculate first-order derivatives of diffraction time with respect to parameters of the time-migration velocity model.

The first-order derivative of diffraction time with respect to a parameter ν of the time-migration velocity model is given by

$$\frac{\partial T^D}{\partial \nu} = \frac{\partial T^S}{\partial \nu} + \frac{\partial T^R}{\partial \nu}. \quad (\text{B.1})$$

Since the derivatives of T^S and T^R have the same form, we only specify derivatives with respect to T^S in the following.

Consider a model parameter $\nu = \nu^{(2)}$ associated with the diffraction-time coefficient $S_{IJ}^{M(2)}$. For this parameter differentiation of the first formula of Eq. (7) yields

$$\frac{\partial T^S}{\partial \nu^{(2)}} = \frac{1}{2T^S} \frac{\partial}{\partial \nu^{(2)}} \left(S_{IJ}^{M(2)} \right) h_I^S h_J^S. \quad (\text{B.2})$$

Likewise, for a model parameter $\nu = \nu^{(4)}$ associated with the diffraction-time coefficient $S_{IJKL}^{M(4)}$ we obtain

$$\frac{\partial T^S}{\partial \nu^{(4)}} = \frac{1}{2T^S} \frac{\partial}{\partial \nu^{(4)}} \left(S_{IJKL}^{M(2)} \right) h_I^S h_J^S h_K^S h_L^S. \quad (\text{B.3})$$

B.1. Derivatives of one-way time for rotationally symmetric diffraction moveout

In the special case where the diffraction-time moveout is rotationally symmetric, Eqs (B.2) and (B.3) are simplified to

$$\frac{\partial T^S}{\partial \nu^{(2)}} = \frac{1}{2T^S} \frac{\partial}{\partial \nu^{(2)}} \left(S^{M(2)} \right) \left(h^S \right)^2, \quad (\text{B.4})$$

$$\frac{\partial T^S}{\partial \nu^{(4)}} = \frac{1}{2T^S} \frac{\partial}{\partial \nu^{(4)}} \left(S^{M(4)} \right) \left(h^S \right)^4. \quad (\text{B.5})$$

APPENDIX C REPRESENTATION OF A TRIVARIATE LOCAL FUNCTION IN TERMS OF B-SPLINE BASIS FUNCTIONS

We consider the representation of a trivariate function in terms of "B-spline" basis functions. The generality is trivially extended to yield a multicomponent function and a number of variables higher than three.

C.1. A hierarchy of local functions for a grid cell

Without loss of generality, consider the grid cell for which $i=j=k=1$. The normalized coordinates for this grid cell shall be denoted as u , v , and w . Each of these coordinates extends between 0 and 1 inside the cell. Generalizing the results of *Bartels et al. (1987)*, we write the approximate value of a function \mathcal{M} in a point inside the cell as

$$\mathcal{M}(u, v, w) = \sum_{k=k_{\min}}^{k=k_{\max}} b_k(w) \sum_{j=j_{\min}}^{j=j_{\max}} b_j(v) \sum_{i=i_{\min}}^{i=i_{\max}} b_i(u) M^{i,j,k}. \quad (\text{C.1})$$

Here, $M^{i,j,k}$ is a known value of the global function $M(\xi)$ in the grid point $X^{i,j,k}$, and the functions $b_i(u)$, $i = i_{\min}, \dots, i_{\max}$ are referred to as basis functions. Equation (C.1) may be used for all three local function representations discussed in Section 2.5. The basis functions and/or the limits of the sums in Eq. (C.1) are however different for a locally constant function, a local trilinear interpolator, and a local tricubic spline function.

Assume now that an unknown parameter v to be estimated by the iterative linearized process is a coefficient of the type $M^{r,s,t}$ included on the right-hand side of Eq. (C.1). In order to obtain Fréchet derivatives, we may need, as indicated by Eq. (56), the derivative $\partial T^D / \partial v$ and therefore also $\partial \mathcal{M} / \partial v$. Since Eq. (C.1) is linear in the coefficients $M^{i,j,k}$, we obtain

$$\frac{\partial \mathcal{M}}{\partial M^{r,s,t}}(u, v, w) = b_i(u) b_j(v) b_k(w) \delta_{ir} \delta_{js} \delta_{kt}.$$

Consequently we have

$$\frac{\partial \mathcal{M}}{\partial M^{r,s,t}}(u, v, w) = \begin{cases} b_r(u) b_s(v) b_t(w) \\ \text{if } r \in [i_{\min}, i_{\max}], s \in [j_{\min}, j_{\max}], t \in [k_{\min}, k_{\max}], \\ 0 \text{ otherwise.} \end{cases} \quad (\text{C.2})$$

Equation (C.2) is interesting in several respects. In particular, it shows that derivatives of the type $\partial \mathcal{M}_\lambda / \partial v$ are equal for functions \mathcal{M}_λ that are evaluated in the same location and are defined with respect to the same grid. Careful implementation of Eq. (C.2) is therefore expected to be very beneficial with respect to the efficiency of the Fréchet derivative calculations.

C.2. Taking the mean value

Taking the summation limits in Eq. (C.1) as $i_{\min} = j_{\min} = k_{\min} = 0$ and $i_{\max} = j_{\max} = k_{\max} = 1$, and defining the basis functions as

$$b_0(u) = \frac{1}{2}, \quad b_1(u) = \frac{1}{2}, \quad (\text{C.3})$$

equation (C.1) is reduced to

$$\mathcal{M}(u, v, w) = \frac{1}{8} \sum_{i=0}^1 \sum_{j=0}^1 \sum_{k=0}^1 M^{i,j,k}. \quad (\text{C.4})$$

In other words, the constant local function may be obtained by taking the mean value of the values $M^{i,j,k}$ in the vertices bounding the grid cell. Moreover, Eq. (C.2) gets the simple form

$$\frac{\partial \mathcal{M}}{\partial M^{r,s,t}}(u, v, w) = \begin{cases} \frac{1}{8} & \text{if } r \in [0, 1], s \in [0, 1], t \in [0, 1], \\ 0 & \text{otherwise.} \end{cases}$$

C.3. Trilinear interpolation

For trilinear interpolation we have the summation limits $i_{\min} = j_{\min} = k_{\min} = 0$ and $i_{\max} = j_{\max} = k_{\max} = 1$, and the basis functions are given by

$$b_0(u) = 1 - u, \quad b_1(u) = u. \quad (\text{C.6})$$

Writing out the various terms in Eq. (C.1) we therefore obtain

$$\begin{aligned}\mathcal{M}(u, v, w) = & b_0(u)b_0(v)b_0(w)M^{000} + b_1(u)b_0(v)b_0(w)M^{100} \\ & + b_0(u)b_1(v)b_0(w)M^{010} + b_1(u)b_1(v)b_0(w)M^{110} \\ & + b_0(u)b_0(v)b_1(w)M^{001} + b_1(u)b_0(v)b_1(w)M^{101} \\ & + b_0(u)b_1(v)b_1(w)M^{011} + b_1(u)b_1(v)b_1(w)M^{111}.\end{aligned}\quad (\text{C.7})$$

C.4. Tricubic spline evaluation

Considering tricubic spline evaluation, the summation limits in Eq. (C.1) become $i_{\min} = j_{\min} = k_{\min} = -1$ and $i_{\max} = j_{\max} = k_{\max} = 2$, and the basis functions are given by *Bartels et al. (1987)*,

$$\begin{aligned}b_{-1}(u) &= \frac{1}{6}(1 - 3u + 3u^2 - u^3), \quad b_0(u) = \frac{1}{6}(4 - 6u^2 + 3u^3), \\ b_1(u) &= \frac{1}{6}(1 + 3u + 3u^2 - 3u^3), \quad b_2(u) = \frac{1}{6}u^3.\end{aligned}\quad (\text{C.8})$$

APPENDIX D CURVATURE ESTIMATION BY QUADRATIC GRADIENT STRUCTURE TENSOR (QST)

The quadratic gradient structure tensor (QST) (*van de Weijer et al., 2001; Bakker, 2002*) is a method for estimating local curvature. The main assumption behind this method is that we are considering a locally quadratic surface on the form:

$$S(\mathbf{x}) \approx \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b} \mathbf{x} + c = 0, \quad (\text{D.1})$$

where \mathbf{A} is a symmetric 3×3 matrix with at least two non-zero eigenvalues and \mathbf{b} is the unit normal vector to the surface. When observing the surface in the reflector-oriented coordinate system (u, v, w) (the u axis is normal to the surface), the surface becomes:

$$S(\mathbf{x}) \approx \frac{1}{2}\kappa_1 v^2 + \frac{1}{2}\kappa_2 w^2 + u, \quad (\text{D.2})$$

where κ_1 and κ_2 are the principal curvatures of the surface. Further, it is shown in *Bakker (2002)* that these curvatures can be estimated by considering the transform that deforms S into a plane. This results in the expressions for obtaining the curvatures in the reflector-oriented coordinate system:

$$\kappa_1 = \frac{v g_u g_v}{v^2 g_u^2}, \quad (\text{D.3})$$

$$\kappa_2 = \frac{w g_u g_w}{w^2 g_u^2}, \quad (\text{D.4})$$

where g_u , g_v and g_w are the gradients along the axes in the reflector-oriented coordinate system.

In *Bakker (2002)*, the vectors spanning this coordinate system are given by the eigenvectors of the GST:

$$\mathbf{u} = [x_u, y_u, t_u]^T \triangleq \mathbf{v}_1, \quad \mathbf{v} = [x_v, y_v, t_v]^T \triangleq \mathbf{v}_2, \quad \mathbf{w} = [x_w, y_w, t_w]^T \triangleq \mathbf{v}_3. \quad (\text{D.5})$$

In that case, the extracted curvatures will be the principal curvatures of the surface. We are, however, seeking the derivatives with respect to x and y . Therefore, we do not use the eigenvectors to span this coordinate system but force \mathbf{v} and \mathbf{w} to be oriented along the x and y axes.

Further more, *Bakker (2002)* shows that estimates of the second order derivatives are obtained by accounting for the rotation of the reflector-oriented coordinate system. Assuming that the two approaches for defining the reflector-oriented coordinate system coincide (that the principal curvature is the same as the derivative with respect to x), the curvatures (estimates of the second order derivatives) are given as:

$$\kappa_x = \frac{\widehat{\partial t^2}}{\partial^2 x} = \kappa_1 \left[1 + \left(\frac{\partial t}{\partial x} \right)^2 \right]^{3/2}, \quad (\text{D.6})$$

$$\kappa_y = \frac{\widehat{\partial t^2}}{\partial^2 y} = \kappa_2 \left[1 + \left(\frac{\partial t}{\partial y} \right)^2 \right]^{3/2}. \quad (\text{D.7})$$

APPENDIX E TIME-MIGRATION VELOCITY MODEL DEFINITION

We define the time-migration model on a 3D rectangular grid. The model parameters related to the cells and the corresponding vertices within the grid. As shown Fig. E1, a rectangular grid cell (i, j, k) is defined in terms of eight vertices $X_{i-1, j-1, k-1}$, $X_{i, j-1, k-1}$, $X_{i-1, j, k-1}$, $X_{i, j, k-1}$, $X_{i-1, j-1, k}$, $X_{i, j-1, k}$, $X_{i-1, j, k}$, $X_{i, j, k}$. We assume that the values of the function M_λ is known in all relevant grid vertices $X_{i, j, k}$, and we denote these values as $M_\lambda^{i, j, k}$.

As in (depth) tomography, we apply here a local function $\mathcal{M}_\lambda(u, v, w)$, which pertains only to one selected cell. The variables u, v, w are dimensionless and take values on the interval $[0, 1]$. At any point within the cell the global and local functions must yield the same output value,

$$M_\lambda(\xi_1, \xi_2, \xi_3) = \mathcal{M}_\lambda(u, v, w). \quad (\text{E.1})$$

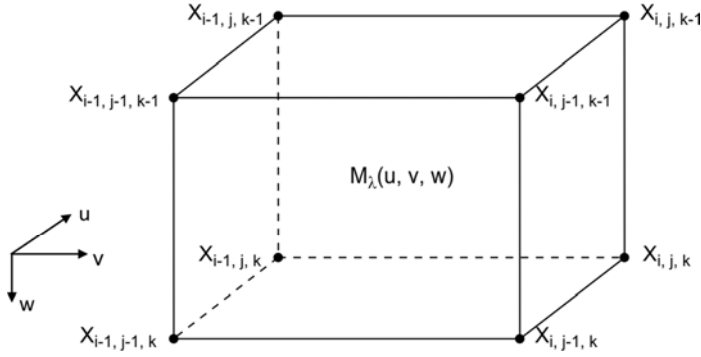


Fig. E1. Grid cell and local dimensionless coordinates (u, v, w) used for describing the 3D time-migration velocity model.

The evaluation of the function \mathcal{M}_λ is based on the values $M_\lambda^{i,j,k}$ in the vertices forming the cell, and eventually also on values in grid vertices that are neighbors to the cell. In the following, we consider three ways of representing the local function \mathcal{M}_λ in a specific grid cell (i, j, k) .

E.1. Constant local function

A common approach in tomography is to assume that the value of the function \mathcal{M}_λ is constant within each grid cell. Using this approach, the medium within each cell is treated as homogeneous. Application of constant local functions will give rise to discontinuities in the function values across grid cell boundaries.

E.2. Linear interpolation

Depending on whether the grid is 1D, 2D, or 3D, one may use linear, bilinear, or trilinear interpolation in order to obtain function values inside a grid cell. These interpolation approaches yield a continuous function value along and across the grid cell boundaries, while the first- and second-order derivatives with respect to the coordinate x_i are generally discontinuous. The number of grid vertices contributing to linear, bilinear, and trilinear interpolation are, respectively, two, four, and eight.

E.3. Local spline approximation

In order to achieve continuity of zeroth-, first- and second-order derivatives, one may use a local spline function. For 1D, 2D, and 3D grids the spline function is referred to as cubic, bicubic and tricubic, respectively. The spline function does not interpolate exactly, meaning that the function value \mathcal{M}_λ in the grid vertex $X_{i,j,k}$ is not necessarily equal to the input value $M_\lambda^{i,j,k}$. However, the local spline function possesses the so-called convex

hull property, which prevents the function from fluctuating drastically between the grid vertices. As approximate coefficients of the function we take the input values to function generation given in the grid vertices. For cubic, bicubic, and tricubic spline evaluation, the number of contributing coefficients are, respectively, 4, 16, and 64.

APPENDIX F PARTIAL DERIVATIVES RELATED TO THE TIME-MIGRATION VELOCITY MODEL

Our time-migration velocity model is described in terms of the vector function M_λ , $\lambda = 1, \dots, N^\lambda$. Within a given grid cell under consideration, the global function M_λ and the corresponding local function \mathcal{M}_λ for the grid cell must be equal. Using Eq. (C.1) we can then write

$$M_\lambda(m_1, m_2, \tau) = \mathcal{M}_\lambda(u, v, w) = \sum_{k=k_{\min}}^{k_{\max}} b_k(w) \sum_{j=j_{\min}}^{j_{\max}} b_j(v) \sum_{i=i_{\min}}^{i_{\max}} b_i(u) M_\lambda^{i,j,k}. \quad (\text{F.1})$$

When differentiating the function M_λ with respect to the variables m_1 , m_2 , and τ , we can utilize their one-to-one linear relationship with the corresponding dimensionless grid-cell variable,

$$m_1 \leftrightarrow u, \quad m_2 \leftrightarrow v, \quad \tau \leftrightarrow w. \quad (\text{F.2})$$

For example, differentiation of Eq. (F.1) with respect to τ yields

$$\frac{\partial M_\lambda}{\partial \tau}(m_1, m_2, \tau) = \frac{1}{\Delta \tau} \sum_{k=k_{\min}}^{k_{\max}} \frac{db_k}{dw}(w) \sum_{j=j_{\min}}^{j_{\max}} b_j(v) \sum_{i=i_{\min}}^{i_{\max}} b_i(u) M_\lambda^{i,j,k}, \quad (\text{F.3})$$

where $\Delta \tau$ signifies the grid spacing in the τ direction. Likewise, a partial second derivative in, say, m_1 and τ is given by

$$\frac{\partial^2 M_\lambda}{\partial m_1 \partial \tau}(m_1, m_2, \tau) = \frac{1}{\Delta m_1 \Delta \tau} \sum_{k=k_{\min}}^{k_{\max}} \frac{db_k}{dw}(w) \sum_{j=j_{\min}}^{j_{\max}} b_j(v) \sum_{i=i_{\min}}^{i_{\max}} \frac{db_i}{du}(u) M_\lambda^{i,j,k}. \quad (\text{F.4})$$

Equations (53) and (56) contain partial derivatives of the diffraction time T^D with respect to a given parameter of the time-migration velocity model, ν . Furthermore, the dependence of T^D on ν is channelled through the diffraction time coefficients M_λ . Now, let the parameter ν be defined as in Eq. (43). Differentiation of Eq. (F.1) with respect to ν yields, in view of Eq. (C.2),

$$\frac{\partial M_\lambda}{\partial \nu}(m_1, m_2, \tau) = \begin{cases} b_r(u) b_s(v) b_t(w) & \text{if } r \in [i_{\min}, i_{\max}]; s \in [j_{\min}, j_{\max}], t \in [k_{\min}, k_{\max}], \\ 0 & \text{otherwise.} \end{cases} \quad (\text{F.5})$$

Equation (56) implies also computation of the second-order mixed partial derivative in the model parameter ν and the time variable τ . Differentiation of Eq. (F.5) with respect to τ yields

$$\frac{\partial^2 M_\lambda}{\partial \nu \partial \tau}(m_1, m_2, \tau) = \begin{cases} \frac{1}{\Delta \tau} b_r(u) b_s(v) \frac{db_t}{dw} b_t(w) \\ \quad \text{if } r \in [i_{\min}, i_{\max}]; s \in [j_{\min}, j_{\max}], t \in [k_{\min}, k_{\max}], \\ 0 \quad \text{otherwise.} \end{cases} \quad (\text{F.6})$$

Acknowledgments: The authors acknowledge support from the Norwegian Research Council through the Petro-Maks 2 project (NFR/234019). We thank Lundin Norway AS for making the 3D field dataset available for this study. We also grateful to NORSAR for providing the software for the synthetic data generation.

References

- Adler F., Baina R., Soudani M.A., Cardon P. and Richard J.-B., 2008. Nonlinear 3d tomographic least-squares inversion of residual moveout in kirchhoff prestack-depth-migration common-image gathers. *Geophysics*, **73**, VE13–VE23.
- Aster R.C., Borchers B. and Thurber C.H., 2013. *Parameter Estimation and Inverse Problems*. 2nd Edition. Academic Press, Waltham, MA.
- Bakker P., 2002. *Image Structure Analysis for Seismic Interpretation*. Ph.D. Thesis. Delft Technical University, Delft, The Netherlands.
- Barros T., Ferrari R., Krummenauer R. and Lopes R., 2015. Differential evolution-based optimization procedure for automatic estimation of the common-reflection surface traveltimes parameters. *Geophysics*, **80**, WD189–WD200.
- Bartels R.H., Beatty J.C. and Barsky B.A., 1987. *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling*. Morgan Kaufmann Publishers, Burlington, MA.
- Bauer A., Schwarz B. and Gajewski D., 2009. Enhancement of prestack diffraction data and attributes using a traveltimes decomposition approach. *Stud. Geophys. Geod.*, **60**, 471–486.
- Baykulov M. and Gajewski D., 2009. Prestack seismic data enhancement with partial common-reflection-surface (CRS) stack. *Geophysics*, **74**, V49–V58.
- Bergler S., Hubral P., Marchetti P., Cristini A. and Cardone G., 2002. 3D common-reflection-surface stack and kinematic wavefield attributes. *The Leading Edge*, **21**, 1010–1015.
- Berkovitch A., Belfer I. and Landa E., 2008. Multifocusing as a method of improving subsurface imaging. *The Leading Edge*, **27**, 250–256.
- Berkovitch A., Belfer I., Hassin Y. and Landa E., 2009. Diffraction imaging by multifocusing. *Geophysics*, **74**, WCA75–WCA81.
- Bigun J. and Granlund G.H., 1987. Optimal orientation detection of linear symmetry. In: *Proceedings of the IEEE First International Conference on Computer Vision*. Computer Society Press of the IEEE, 433–438.
- Billette F. and Lambaré G., 1998. Velocity macro-model estimation from seismic reflection data by stereotomography *Geophys. J. Int.*, **135**, 671–690.

- Blout R., Coimbra T.A., Facciopieri J.H. and Tygel M., 2018. Common-reflection-surface method in weakly anisotropic vertical transverse isotropic media. *Geophysics*, **83**, C99–C113.
- Bóna A., 2011. Shot-gather time migration of planar reflectors without velocity model. *Geophysics*, **76**, S93–S101.
- Bonomi E., Cristini M., Theis D. and Marchetti P., 2009. 3D CRS analysis: a data-driven optimization for the simultaneous estimate of the eight parameters. *SEG Technical Program Expanded Abstracts 2009*, 3284–3291, DOI: 10.1190/1.3255542.
- Chauris H., Noble M.S., Lambaré G. and Podvin P., 2002a. Migration velocity analysis from locally coherent events in 2-d laterally heterogeneous media, part I: Theoretical aspects. *Geophysics*, **67**, 1202–1212.
- Chauris H., Noble M.S., Lambaré G. and Podvin P., 2002b. Migration velocity analysis from locally coherent events in 2-d laterally heterogeneous media, part II: Applications on synthetic and real data. *Geophysics*, **67**, 1213–1224.
- Coimbra T.A., de Figueiredo J.J.S., Schleicher J., Novais A. and Costa J.C., 2013. Migration velocity analysis using residual diffraction moveout in the poststack depth domain. *Geophysics*, **78**, S125–S135.
- Coimbra T.A., Facciopieri J.H., Rueda D.S. and Tygel M., 2016a. Common-reflection-point time migration. *Stud. Geophys. Geod.*, **60**, 500–530.
- Coimbra T.A., Novais A. and Schleicher J., 2016b. Offset-continuation stacking: Theory and proof of concept. *Geophysics*, **81**, V387–V401.
- Cooke D., Bóna A. and Hansen B., 2009. Simultaneous time imaging, velocity estimation, and multiple suppression using local event slopes. *Geophysics*, **74**, WCA65–WCA73.
- Dell S. and Gajewski D., 2011. Common-reflection-surface-based workflow for diffraction imaging. *Geophysics*, **76**, S187–S195.
- Dell S., Gajewski D. and Tygel M., 2014. Image-ray tomography. *Geophys. Prospect.*, **62**, 413–426.
- Dix C.H., 1955. Seismic velocities from surface measurements. *Geophysics*, **20**, 68–86.
- Douma H. and de Hoop M.V., 2006. Leading-order seismic imaging using curvelets. *SEG Technical Program Expanded Abstracts 2006*, 2411–2415, DOI: 10.1190/1.2370019.
- Facciopieri J.H., Serrano D.R., Gelius L.J. and Tygel M., 2013. Recovering diffractions in CRS stacked sections. *First Break*, **31**, 27–31.
- Facciopieri J.H., Coimbra T.A., Gelius L.J. and Tygel M., 2016. Stacking apertures and estimation strategies for reflection and diffraction enhancement. *Geophysics*, **81**, V271–V282.
- Fomel S., 2002. Applications of plane-wave destruction filters. *Geophysics*, **67**, 1946–1960.
- Fomel S., 2007a. Velocity-independent time-domain seismic imaging using local event slopes. *Geophysics*, **72**, S139–S147.
- Fomel S., 2007b. Local seismic attributes. *Geophysics*, **72**, A29–A33.
- Fomel S. and Kazinnik R., 2012. Non-hyperbolic common reflection surface. *Geophys. Prospect.*, **61**, 21–27.
- Garabito G., Cruz J.C.R. and Solner W., 2017. Finite-offset common reflection surface stack using global optimisation for parameter estimation: a land data example. *Geophys. Prospect.*, **65**, 1123–1137.
- Gelius L.-J. and Tygel M., 2015. Migration-velocity building in time and depth from 3D (2D) Common-Reflection-Surface (CRS) stacking - theoretical framework. *Stud. Geophys. Geod.*, **59**, 253–282.

- Gjøystdal H. and Ursin B., 1981. Inversion of reflection times in three dimensions. *Geophysics*, **46**, 972–983.
- Guillaume P., Audebert F., Berthet P., David B., Herrenschmidt A. and Zhang X., 2001. 3D finite-offset tomographic inversion of CRP-scan data, with or without anisotropy. *SEG Technical Program Expanded Abstracts 2001*, 718–721, DOI: 10.1190/1.1816731.
- Guillaume P., Lambaré G., Leblanc O., Mitouard P., Moigne J.L., Montel J.-P., Prescott T., Siliqi R., Vidal N., Zhang X. and Zimine S., 2008. Kinematic invariants: an efficient and flexible approach for velocity model building. *SEG Technical Program Expanded Abstracts 2008*, 3687–3692, DOI: 10.1190/1.3064100.
- Guillaume P., Reinier M., Lambaré G., Cavalié A., Adamsen M.I. and Bruun B.M., 2013. Dip constrained non-linear slope tomography: an application to shallow channels characterization. *SEG Global Meeting Abstracts*, 1587–159, DOI: 10.1190/sbgf2013-325.
- Hertweck T., Schleicher J. and Mann J., 2007. Data stacking beyond CMP. *The Leading Edge*, **26**, 818–827.
- Hoecht G., Ricarte P., Bergler S. and Landa E., 2009. Operator-oriented CRS interpolation. *Geophys. Prospect.*, **57**, 957–979.
- Hubral P., 1999. Special issue: Macro model independent seismic imaging. *J. Appl. Geophys.*, **42**, 137–138.
- Hubral P. and Krey T., 1980. *Interval Velocities from Seismic Reflection Time Measurements*. Society of Exploration Geophysicists, Tulsa, USA.
- Iversen E., 2004. The isochron ray in seismic modeling and imaging. *Geophysics*, **69**, 1053–1070.
- Iversen E. and Gjøystdal H., 1996. Event-oriented velocity estimation based on prestack data in time or depth domain. *Geophys. Prospect.*, **44**, 643–686.
- Iversen E., Tygel M., Ursin B. and Hoop M.V., 2012. Kinematic time migration and demigration of reflections in pre-stack seismic data. *Geophys. J. Int.*, **189**, 1635–1666.
- Jäger R., Mann J., Höcht G. and Hubral P., 2001. Common-reflection-surface stack: Image and attributes. *Geophysics*, **66**, 97–109.
- Khoshnavaz M.J., 2017. Oriented time-domain dip moveout correction for planar reflectors in common-source domain. *Geophysics*, **82**, U87–U97.
- Khoshnavaz M.J., Bóna A., Dzunic A., Ung K. and Urosevic M., 2016a. Oriented prestack time migration using local slopes and predictive painting in the common-source domain for planar reflectors. *Geophysics*, **81**, S409–S418.
- Khoshnavaz M.J., Bóna A. and Urosevic M., 2016b. Velocity-independent estimation of kinematic attributes in vertical transverse isotropy media using local slopes and predictive painting. *Geophysics*, **81**, U73–U85.
- Kleyn A.H., 1977. On the migration of reflection time contour maps. *Geophys. Prospect.*, **25**, 125–140.
- Klokov A. and Fomel S., 2012. Separation and imaging of seismic diffractions using migrated dip-angle gathers. *Geophysics*, **77**, S131–S143.
- Knutsson H., 1989. Representing local structure using tensors. In: Pietikainen M. and Rönig J. (Eds), *Proceedings of the 6th Scandinavian Conference on Image Analysis*. International Association for Pattern Recognition, 244–251.
- Lambaré G., Deladerrière N., Traonmilin Y., Touré J.P., Moigne J.L. and Herrmann P., 2009. Non-linear tomography for time imaging. Extended Abstract. *71st EAGE Conference and Exhibition Incorporating SPE EUROPEC 2009*. EAGE Publications BV, DOI: 10.3997/2214-4609.201400386.

- Landa E., 2007. *Beyond Conventional Seismic Imaging*. EAGE, The Hague, The Netherlands.
- Landa E., Gurevich B., Keydar S. and Trachtman P., 1999. Application of multifocusing method for subsurface imaging. *J. Appl. Geophys.*, **42**, 283–300.
- Landa E., Keydar S. and Moser T.J., 2010. Multifocusing revisited - inhomogeneous media and curved interfaces. *Geophys. Prospect.*, **58**, 925–938.
- Messud J., Lambaré G., Guillaume P. and Rohel C., 2015. Non-linear slope tomography for orthorhombic pre-stack time imaging. Extended Abstract. *77th EAGE Conference and Exhibition 2015*. EAGE Publications BV, DOI: 10.3997/2214-4609.201412575.
- Minato S., Tsuji T., Matsuoka T., Nishizaka N. and Ikeda M., 2012. Global optimisation by simulated annealing for common reflection surface stacking and its application to low-fold marine data in southwest japan. *Explor. Geophys.*, **43**, 59–69.
- Neidell N.S. and Taner M.T., 1971. Semblance and other coherency measures for multichannel data. *Geophysics*, **36**, 482–497.
- Perroud M., Hubral P. and Höcht G., 1999. Common-reflection-point stacking in laterally inhomogeneous media. *Geophys. Prospect.*, **47**, 1–24.
- Rad P.B., Schwarz B., Gajewski D. and Vanelle C., 2018. Common-reflection-surface-based prestack diffraction separation and imaging. *Geophysics*, **83**, S47–S55.
- Randen T., Monsen E., Signer C., Abrahamsen A., Hansen J.O., Sæter T. and Schlaf J., 2000. Three-dimensional texture attributes for seismic data analysis. *SEG Technical Program Expanded Abstracts 2000*, 668–671, DOI: 10.1190/1.1816155.
- Riabinkin 1957. Fundamentals of resolving power of controlled directional reception (CDR) of seismic waves. In: Gardner G.H.F. and Lu L. (Eds), *Slant-Stack Processing*, **14**, 36–60, Society of Exploration Geophysicists, Tulsa, USA.
- Rieber F., 1936. Visual presentation of elastic wave patterns under various structural conditions. *Geophysics*, **1**, 196–218.
- Shah P.M., 1973. Use of wavefront curvature to relate seismic data with subsurface parameters. *Geophysics*, **38**, 812–825.
- Söllner W. and Andersen E., 2005. Kinematic time migration and demigration in a 3D visualization system. *J. Seism. Explor.*, **14**, 255–270.
- Söllner W., Andersen E. and Lima J., 2004. Fast time-to-depth mapping by first-order ray transformation in a 3-D visualization system. *SEG Technical Program Expanded Abstracts 2004*, 1081–1084, DOI: 10.1190/1.1851071.
- Spinner M. and Mann J., 2006. True-amplitude based Kirchhoff time migration for AVO/AVA analysis. *Geophysics*, **15**, 133–152.
- Stolk C.C., de Hoop M.V. and Symes W.W., 2009. Kinematics of shot-geophone migration. *Geophysics*, **74**, WCA19–WCA34.
- Stovas A. and Fomel S., 2015. Mapping of moveout attributes using local slopes. *Geophys. Prospect.*, **64**, 31–37.
- Sword C.H., 1986. Tomographic determination of interval velocities from picked reflection seismic data. *SEG Technical Program Expanded Abstracts 1986*, 657–660, DOI: 10.1190/1.1892933.
- Tygel M. and Santos L.T., 2007. Quadratic normal moveouts of symmetric reflections in elastic media: A quick tutorial. *Stud. Geophys. Geod.*, **51**, 185–206.
- Ursin P., 1982. Quadratic wavefront and travelttime approximations in inhomogeneous layered media with curved interfaces *Geophysics*, **47**, 1012–1021.

- van de Weijer J., van Vliet L., Verbeek P. and van Ginkel R., 2001. Curvature estimation in oriented patterns using curvilinear models applied to gradient vector fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, **23**, 1035–1042.
- Vanelle C., Abakumov I. and Gajewski D., 2018. Wavefront attributes in anisotropic media. *Geophys. J. Int.*, **214**, 430–443.
- Walda J. and Gajewski D., 2017. Determination of wavefront attributes by differential evolution in the presence of conflicting dips. *Geophysics*, **82**, V229–V239.
- Waldeland A.U., Coimbra T.A., Faccipieri J.H., Solberg A.H.S. and Gelius L.J., 2019. Fast estimation of prestack common reflection surface parameters. *Geophys. Prospect.*, **67**, 1163–1183.
- Whitcombe D.N., Murray E.H., Aubin L.A.S. and Carroll R.J., 1994. The application of 3-D depth migration to the development of an alaskan offshore oil field. *Geophysics*, **59**, 1551–1560.
- Yuan S., Wang S., Luo C. and Wang T., 2018. Inversion-based 3-D seismic denoising for exploring spatial edges and spatio-temporal signal redundancy. *IEEE Geosci. Remote Sens. Lett.*, **15**, 1682–1686.
- Yuan S., Su Y., Wang T., Wang J. and Wang S., 2019. Geosteering phase attributes: A new detector for the discontinuities of seismic images. *IEEE Geosci. Remote Sens. Lett.*, **16**, 145–149.
- Zhang Y., Bergler S., Tygel M. and Hubral P., 2002. Model-independent travel-time attributes for 2-D, finite-offset multicoverage reflections. *Pure Appl. Geophys.*, **159**, 1601–1616.