Enhanced constrained velocity inversion

Introduction

Inverting rms velocity functions to generate instantaneous velocities is only one step of the complex velocity model building process. As the velocity model sits at the very beginning of the workflow leading from seismic data to reservoir modelling, errors or inaccuracies can result in misleading reservoir estimations. Hajnal and Sereda (1981) show that the Dix equation (Dix, 1955) routinely used to estimate interval velocities from measured rms velocities carries an inherent uncertainty which may make inversion results unreliable. Furthermore, many different velocity models may produce seismic images fulfilling the flat image gathers requirements. Reasonable assumptions about the velocity model, incorporated into the inversion process and validated as suggested by Glogovsky et al. (2009), can reduce the field of possible solutions and should thus be considered to improve velocity model building.

Viloria et al. (2009)’s comprehensive approach integrates geological and geophysical information into the velocity model. Even then, rms velocity functions at well locations are transformed to interval velocities using the Dix equation without further refinement, requiring post-processing to calibrate the interval velocity volume to wells. Koren and Ravve (2006) adapt the Dix inversion process to include geological constraints, making the inversion result more accurate. However, some geological information such as punctual interval velocity values or physical bounds are missing in the current framework.

In this paper, we review the constrained inversion procedure, then show how the Discrete Smooth Interpolation (DSI) method (Mallet, 1989; Cognot, 1996; Mallet, 2002), developed to solve various geomodelling problems, can be extended to add new types of physical and geological constraints to the Dix inversion. Finally, we show real inversion problems including some of those constraints.

Constrained Dix inversion

The four-step constrained Dix inversion method presented in Koren and Ravve (2006) begins with computing a background velocity trend for each node of a coarse lateral grid, from an estimation of the asymptotic value towards which velocity in fully compacted sediments tends and from picked rms velocity values of vertical functions within a given range of each coarse grid node. In a second step, an unconstrained velocity inversion computes instantaneous velocities from vertical functions of rms velocities at sparse and irregular lateral locations. This produces a first instantaneous velocity which follows the trend and thus, does not exhibit the noisy and oscillatory behaviour associated with the standard Dix inversion. The third step first converts the result of the unconstrained velocity inversion to rms velocity onto a coarse time grid, then computes a cost function from the rms velocity misfit, the trend misfit and an anti-oscillatory damping term. The unconstrained Dix inversion formula provides an initial guess for instantaneous velocities, which is updated by minimising the cost function in an iterative Newton process. Finally, the instantaneous velocities known at lateral locations of a coarse time grid are transferred to a fine grid by a 2D gridding procedure.

In the third step described above, minimising the cost function to compute velocity updates is a numerical optimisation process in which the problem to be solved can be enriched by additional constraints, using an appropriate framework, such as Discrete Smooth Interpolation (DSI), presented in the next section.

DSI Theory

The DSI problem can be formulated as finding the function $\varphi$ which best honours, in a least squares sense, a series of linear constraints $C$ involving any number of connected nodes in a finite set $\Omega$ of $M$ nodes. This set may represent the discretisation of any type of geological object such as triangulated faults or horizons, different kinds of grids and even purely abstract multi-dimensional objects.

DSI takes into account both soft constraints $C^s$, honoured in a least squares sense (some constraints may be violated more or less significantly to minimise the global residual), and hard equality or inequality
constraints $\mathcal{G}^h$, honoured exactly. If $\varphi$ is an unknown, scalar function defined on set $\Omega$ of $M$ nodes and represented as a column vector, each individual constraint applied to $\varphi$ is of the following form:

$$\forall c_i \in \mathcal{G}^s, a_i^t \cdot \varphi = b_{c_i} \quad \text{and} \quad \forall c_i \in \mathcal{G}^h, \begin{cases} a_i^t \cdot \varphi = b_{c_i} \\ a_i^t \cdot \varphi \geq b_{c_i} \end{cases},$$

(1)

where $a_{c_i}$ is a given column vector with $M$ elements representing the contribution of each individual node of $\Omega$ to constraint $c_i$, some of which may be zero, and $b_{c_i}$ is a given value. Any linear equation combining any nodes in discrete model $\Omega$ can thus be translated into a soft or hard DSI constraint. Solving the soft constraints part of the DSI problem is equivalent to solving the set of $K$ linear equations:

$$\begin{bmatrix} a_{c_1}^t \\ \vdots \\ a_{c_M}^t \end{bmatrix} \cdot \varphi = \begin{bmatrix} b_{c_1} \\ \vdots \\ b_{c_M} \end{bmatrix}, \quad c_i \in \mathcal{G}^s.$$  

(2)

Each constraint $c_i$ in this system is normalised and may be individually weighted in order to increase the impact of some constraints on the solution. Solving system (2) can be expressed as minimising the soft constraints residuals, written as quadratic criterion $Q$:

$$Q(\varphi) = \sum_{c_i \in \mathcal{G}^s} (a_{c_i}^t \cdot \varphi - b_{c_i})^2.$$  

(3)

Developing $(a_{c_i}^t \cdot \varphi - b_{c_i})^2$ yields $(a_{c_i}^t \cdot \varphi - b_{c_i})^2 = \varphi^t \cdot (a_{c_i} \cdot a_{c_i}^t) \cdot \varphi - 2(b_{c_i} \cdot a_{c_i}^t) \cdot \varphi + b_{c_i}^2$.

Defining square symmetric matrix $[A]$ as $[A] = \sum_{c_i \in \mathcal{G}^s} a_{c_i} \cdot a_{c_i}^t$ and column vector $b$ as $b = \sum_{c_i \in \mathcal{G}^s} b_{c_i} \cdot a_{c_i}$, we can reformulate minimisation problem (3) as minimising quadratic criterion $J$:

$$J(\varphi) = \frac{1}{2} \varphi^t \cdot [A] \cdot \varphi - b^t \cdot \varphi.$$  

(4)

a well-known optimisation problem which can be numerically solved in a number of efficient ways.

If set $\mathcal{G}^h$ contains hard constraints defined on nodes of $\Omega$, we arrange the $L$ hard constraints in a $L \times M$ matrix $A_h$ and the $L$ right-hand side values in a vector $b_h$ as follows:

$$A_h = \begin{bmatrix} a_{c_1}^t \\ \vdots \\ a_{c_L}^t \end{bmatrix} \quad \text{and} \quad b_h = \begin{bmatrix} b_{c_1} \\ \vdots \\ b_{c_L} \end{bmatrix}, \quad c_i \in \mathcal{G}^h.$$  

(5)

We then formulate the Lagrangian of the problem:

$$\Lambda(\varphi, \lambda) = J(\varphi) + \lambda^t (A_h \cdot \varphi - b_h),$$

(6)

thus adding one additional variable $\lambda_i$ per hard constraint $c_i \in \mathcal{G}^h$. The solution $\varphi(\lambda)$ to the hard-constrained problem is a saddle point where $\Lambda$ is minimised relative to $\varphi$, $\Lambda$ is maximised relative to $\lambda_i$, and $\forall i, \lambda_i \geq 0$. As a consequence, $\varphi(\lambda)$ is such that $\nabla_\varphi \Lambda(\varphi(\lambda), \lambda) = 0$.

As $\nabla_\varphi \Lambda(\varphi, \lambda) = [A] \cdot \varphi - b + A_h \cdot \lambda$, we get

$$\varphi(\lambda) = [A]^{-1}(b - A_h \cdot \lambda).$$  

(7)

Injecting (7) into (6) gives the form of $\Lambda(\varphi(\lambda), \lambda)$ which we can use to solve the $\lambda_i$ positively constrained Lagrangian of the problem and thus obtain solution $\varphi$ to the DSI problem with hard constraints.

**CVI with added DSI constraints**

Viloria et al. (2009) calibrate an interval velocity volume to wells by applying a correction factor computed from actual interval velocity values derived from well time versus depth information. By adding
a DSI constraint to our Dix inversion method, we use this information directly when computing interval velocities from rms in the vertical functions. Figure 1 shows a vertical function of interval velocities computed from input rms. The blue curve is the result obtained from the current CVI process, using an exponential asymptotic trend (Ravve and Koren, 2006) which tends to a velocity of 5 km/s at large depths. The red curve results from a DSI minimisation of the same problem, using the same data misfit, trend misfit and damping energy as the CVI formulation, with added constraints. First, an inequality constraint is added to ensure interval velocities obtained through DSI do not exceed the 5 km/s cap. As the constrained velocity inversion process minimises a cost function on velocity updates, not velocities themselves, the DSI constraint is expressed in terms of velocity updates $\Delta V$:

$$\forall n \in \{0, N - 1\}, \quad V_{i, n} + \Delta V_{i, n} \leq V_{\text{max}},$$

with $V_{\text{max}}$ set to 5 km/s, where $N$ is the number of nodes in the vertical function and $i$ represents the current iteration of the Newton method. Next, in order to simulate the introduction of additional data into the inversion process, we set two soft constraints (green dots) on the inverted velocity. As these additional data points may not coincide with nodes in the vertical function, we use a linear combination of velocity updates at the closest nodes. Between nodes, velocity is assumed to vary linearly in depth, so exponentially in time, but for this conceptual study we consider the step between grid nodes in time small enough that the error introduced by a linear variation in time between nodes is not significant. Adding an instantaneous velocity data point $V_{0,n,\xi}$ between nodes $n$ and $n + 1$, $\xi \in [0, 1]$ representing the barycentric coordinate of the data point on grid segment $n, n + 1$, is achieved by adding the following DSI constraint to the problem:

$$(V_{0,n} + \Delta V_{0,n}) \cdot (1 - \xi) + (V_{0,n+1} + \Delta V_{0,n+1}) \cdot \xi \simeq V_{0,n,\xi}$$

Finally, an extra regularisation term is added to smooth the variations of instantaneous velocity in the vicinity of constraint points, by specifying that the gradient of the instantaneous velocity must remain approximately constant from one pair of grid nodes to the next.

Solving the CVI with added DSI constraints results in the red curve in Figure 1. The asymptotic trend is honoured and the inequality constraints ensure the maximum velocity of 5 km/s is not exceeded. The added soft constraints, at 2.5 km/s between 0.3 s and 0.35 s, and 2.8 km/s between 0.5 s and 0.55 s, are honoured in a least squares sense. The lower part of Figure 1 is a zoomed-in section of the graph which clearly shows that the DSI interpolation result (red curve) approximates the added constraints (green dots) whilst still preserving the general shape of the CVI result (blue curve).

In a second example shown in Figure 2, we use the internal trend computed in the first step of the current CVI method to set bounds on the velocity updates applied to the first guess for the inversion so that the
final, inverted velocities do not deviate too much from the trend. To illustrate the effects of setting those bounds, we also add soft constraints, some of which with instantaneous velocity values purposely outside of the bounds. If at a given node $n$, the internal trend value is $T(n)$, then the constraints set on updates $\Delta V_{0,n}^i$ at iteration $i$ to ensure the updated inverted velocity function remains within interval $[(1-f)T(n), (1+f)T(n)]$, with $f$ a small value (set to 0.1 in our example), read:

$$(1-f) \cdot T(n) \leq V_{0,n}^i + \Delta V_{0,n}^i \leq (1+f) \cdot T(n)$$ (10)

For each node, these two inequality constraints are recomputed at each step $i$ of the Newton method and ensure the final interval velocity remains within the specified range of the internal trend, even when added soft constraints would tend to make it deviate further. Thus, in Figure 2, added interval velocity constraint A, which is inside the bounds set from the internal trend, is honoured in a least squares sense. However, the upper bound prevents the interpolation result from honouring constraints B and C. In the deeper part, a low velocity anomaly is introduced as a series of seven soft constraints (D). The interpolation result reflects the anomaly but its amplitude is restricted by the lower bound set from the internal trend. This second example uses an internal trend to constrain the velocity inversion but if an external trend function is provided as input to the CVI process, it may be used to constrain the inversion result in a similar way.

**Conclusion**

Using a flexible mathematical tool such as DSI opens new perspectives for CVI. Even though a few approximations were made in order to obtain the first results presented here, ultimately any physical phenomenon or geological attribute which can be translated into a linear equation involving vertical function nodes can be integrated into the existing system and solved, either in a least-squares manner or as a hard constraint. DSI can thus take into account sparse and even conflicting data to give a plausible solution to the inversion problem.

DSI is not limited to 1D interpolation. Currently, velocity inversion is performed on individual vertical functions, with a gridding process then propagating the results to a fine lateral grid. With added lateral continuity constraints, the whole inversion could be performed in three dimensions using DSI. Then, looking beyond CVI, advanced seismic wave and ray based tomographic applications for the determination of subsurface elastic / viscoelastic model parameters (isotropic / anisotropic velocity fields) involve integrating multi-disciplinary datasets: multiple types of seismic data from different acquisition surveys, different characteristics and attributes, well data, potential fields (e.g., EM and gravity), a priori information about geological setting, compaction effects and lateral tectonic stress, etc. Each of these items should be introduced individually into this comprehensive global inversion process, with its own inherent physical constraints. DSI is an attractive mathematical framework for such a challenge.

**References**


