Stanford Exploration Project, Report 61, January 13, 1998, pages 1–32

Velocity analysis by inversion

Josef Jedlička[1]

keywords: *velocity, sampling, stacking*

ABSTRACT

Velocity analysis using a conjugate gradient method gives a better resolution than that obtained by classical stacking along hyperbolas. The data can be easily obtained back from this velocity analysis panel.
Comparisons of various ways of sampling in the velocity domain show that even sampling in slowness squared should be chosen.

INTRODUCTION

The aim of velocity analysis is to get the velocity analysis panel with the best possible resolution. Ideally a hyperbola in a CMP gather should be mapped onto a point in a velocity analysis panel. The operator of summation along hyperbolas does not give this resolution.

If we find such an operator $\mathbf{L}$ that transforms a point in a velocity space into a hyperbola in an offset space, then by solving the equation

$$\mathbf{L}\mathbf{u} = \mathbf{d} \tag{1}$$

we obtain a velocity analysis panel with improved resolution (Thorson, 1984, Thorson and Claerbout, 1985). Those authors use stochastic inversion for solving this problem with a projected gradient descent algorithm.

In this paper, I solve the equation (1) using a conjugate gradient method for minimizing $||\mathbf{L}\mathbf{u} - \mathbf{d}||^2$. Even sampling in the slowness squared (sloth) domain is used. I finally compare various ways of sampling in the velocity domain.

DEFINITION OF OPERATORS

Let us denote $\mathbf{H}$ as the operator of summation along hyperbolas. The operator $\mathbf{H}$ transforms (t, x)-space into (τ, m)-space:

$$\mathbf{u} = \mathbf{H}\mathbf{d}. \tag{2}$$

[1]**email:** not available

An adjoint operator $\mathbf{H^T}$ transforms (τ, m)-space into (t, x)-space:

$$\mathbf{d} = \mathbf{H^T}\mathbf{u}. \tag{3}$$

We can also define a stacking operator in (τ, m)-space. Because $\mathbf{H}$ maps a point onto a parabola (Jedlička, 1989b), it is natural to define an operator $\mathbf{P}$ as an operator of summation along parabolas:

$$\mathbf{d} = \mathbf{P}\mathbf{u}. \tag{4}$$

The adjoint operator $\mathbf{P^T}$,

$$\mathbf{u} = \mathbf{P^T}\mathbf{d}, \tag{5}$$

may serve as an alternative operator for velocity analysis. Here each point in (t, x)-space is spread onto a parabola in (τ, m)-space. Both operators $\mathbf{H}$ and $\mathbf{P}$ are linear. The relationship between them is shown in Appendix A.

VELOCITY ANALYSIS

We know about two operators that map a point in (τ, m)-space onto a hyperbola in (t, x)-space. These operators are $\mathbf{P}$ and $\mathbf{H^T}$. If we sum along parabolas (operator $\mathbf{P}$) passing through one point in (τ, m)-space, we obtain a hyperbola in (t, x)-space. A hyperbola is also obtained when a point in (τ, m)-space is spread onto a hyperbola in (t, x)-space (operator $\mathbf{H^T}$). Therefore, to obtain a velocity analysis panel $\mathbf{u}$, we solve

$$\mathbf{P}\mathbf{u} = \mathbf{d} \tag{6}$$

or

$$\mathbf{H^T}\mathbf{u} = \mathbf{d}. \tag{7}$$

Solving the equation

Equations (6) and (7) usually contain an enormous number of variables and, as will be shown later, are extremely ill-conditioned. However, good results may be obtained in a small number of iterations if a conjugate gradient method is used. To solve the equation

$$\mathbf{L}\mathbf{u} = \mathbf{d} \tag{8}$$

we can minimize the function

$$\mathbf{f} = \|\mathbf{L}\mathbf{u} - \mathbf{d}\|^2, \tag{9}$$

or for stochastic inversion,

$$\mathbf{f} = \frac{\|\mathbf{L}\mathbf{u} - \mathbf{d}\|^2}{\sigma_d^2} + \frac{\|\mathbf{u}\|^2}{\sigma_u^2}, \tag{10}$$

where σ_d^2 and σ_u^2 are the variances of the data and the model spaces respectively.

When a velocity analysis panel is obtained by the solution of equation (6) or equation (7), an inversion is easily done by application of the operator $\mathbf{P}$ or $\mathbf{H^T}$ on the velocity analysis panel $\mathbf{u}$.

We can also consider solving the equation

$$\mathbf{Hd} = \mathbf{u}, \tag{11}$$

where $\mathbf{u}$ was obtained by the application of $\mathbf{H}$ on (t, x)-space, and

$$\mathbf{P^T d} = \mathbf{u}, \tag{12}$$

where $\mathbf{u}$ was obtained by the application of $\mathbf{P^T}$ on (t, x)-space.

COMPUTATION OF ADJOINT OPERATORS

The adjoint operator $\mathbf{H^T}$ to the summation along a hyperbola in (t, x)-space is the spreading of each point in (τ, m)-space onto a hyperbola in (t, x)-space. Similarly, the adjoint operator $\mathbf{P^T}$ to the summation along a parabola in (τ, m)-space is the spreading of each point in (t, x)-space onto a parabola in (τ, m)-space.

The application of $\mathbf{H^T}$ may be summarized as follows:

`for all sloths` m
  `for all offsets` x
    $\mathbf{d}_x = \mathbf{d}_x + \mathbf{NMO^T u}_m$.

$\mathbf{NMO^T}$ is a transpose operation to $\mathbf{NMO}$ (Claerbout, 1985). In words, the process proceeds as follows:

The algorithm takes the first trace of the velocity analysis panel, applies $\mathbf{NMO^T}$ corresponding to the first offset and adds the result to the first offset. Then $\mathbf{NMO^T}$ is applied to the same trace for the second offset and the result is added to the second offset, and so on for other offsets. The same process is repeated with the rest of the traces of the velocity analysis panel.

For the operator $\mathbf{P^T}$ the process is the same; the only difference is that $\mathbf{NMO^T}$ is replaced by the transpose to the corresponding compressing operation.

DATA INVERSION

A common-midpoint gather from Southern California (Figure **??**a) was used to test the algorithm. A velocity panel after one iteration with the operators $\mathbf{P^T}$ and $\mathbf{P}$ is shown in Figure **??**a. This corresponds to the usual velocity stack, but instead of stacking along hyperbolas, stretching along parabolas was used. The velocity analysis panel after ten iterations is shown in Figure ??b. We can see that the resolution has improved and a series of multiples have become apparent. A contour map of both velocity panels is in Figure ??. The resolution is better in Figure ??b, but a better comparison can be seen from Figure ??, which shows the increase of amplitudes in successive iterations.

By applying a transpose operator on the velocity panel we should get back the input data. The results produced by one and ten iterations are shown in Figure ??b and Figure ??a. We can see that amplitudes decrease with offset after one iteration. The explanation is in the geometrical representation of $\mathbf{H^T H}$. (The explanation should be for the operator $\mathbf{PP^T}$ in this case.) The error after ten iterations is shown in Figure ??b.

We can perform velocity analysis for various ways of sampling in the velocity space. We should choose the sampling for which the convergence is the fastest. The comparison of least-squares errors for even sampling in velocity, slowness, and sloth domains is shown in Figure ??a and Figure ??b. We can see that the least errors are in the sloth domain. Convergence is faster with operators $(\mathbf{H}, \mathbf{H^T})$ than with $(\mathbf{P^T}, \mathbf{P})$ for a small number of iterations.

If the lower limit of slowness is chosen to be greater than zero, convergence becomes faster. We will have a look at the cause of this effect later. It appears that even sampling in the velocity domain should not be chosen.

Modifications of basic equations

We can see that when we want to get back the initial data from the velocity panel, amplitudes at far offsets tend to resume their original values in a higher number of iterations than amplitudes at near offsets. This suggests that weighting depending on offset could speed the iterations.

Claerbout (1987) uses a weight that contains a square root of offset. It contains other factors such as time, slowness and count. Count does not play a significant role in most of the gather (see the section in this paper on count). I obtained the best results with this kind of weight when I used a square root of offset as weight. From Figure ??a we can see that this weight leads to the attenuation of near offsets. The result after one iteration may seem to look better with weighting than without weighting, but other iterations converge more slowly. Incorporating other factors into the weight led to the attenuation of amplitudes at high times.

Another possibility of how to give a higher weight to the far offsets is to stack all the amplitudes between two close hyperbolas. This is similar to the stochastic normal moveout (SMOC) (Jedlička, 1989a), but there is no normalizing as in SMOC. It probably corresponds to the "antialias" operator in Claerbout (1987). There is a parameter describing how close the hyperbolas are. By playing with this parameter I obtained a result after one iteration (Figure ??b) with the same strength of ampli-

Operator in		Dot product	
time domain	sloth domain	$(\mathbf{L^T d}, \mathbf{L^T d})$	$(\mathbf{L L^T d}, \mathbf{d})$
$\mathbf{H}$	$\mathbf{H^T}$	110.5207	110.5218
$\mathbf{P^T}$	$\mathbf{P}$	106.6519	106.6530
$\sqrt{\tau/t}\,\mathbf{H}$	$\sqrt{t/\tau}\,\mathbf{P}$	107.6680	108.4489
$\mathbf{H}$	$t/\tau\,\mathbf{P}$	110.5207	111.3971
$\tau/t\,\mathbf{H}$	$\mathbf{P}$	105.1228	105.8237
$\mathbf{H}$	$\mathbf{P}$	110.5207	108.4480

Table 0.1: Dot product test for various operators.

tudes over all the offsets. Other iterations again converge more slowly and an event resembling a direct arrival appeared (Figure **??**a). The velocity of the event is equal to the lowest velocity used in the inversion, which suggests that we have something to do with a computational artifact. A closer inspection of this event is in the section on count (Figure **??**). From this it follows that muting in NMO should attenuate this effect (Figure **??**b).

Weighted modifications of the equations may serve for a quick look at the data after one iteration. For more iterations non-weighted equations give better results.

Dot product test

In Appendix A transpose operators to hyperbola and parabola stacking in continuous domain are derived. For hyperbola stacking $\mathbf{H}$ it is weighted parabola stacking in (τ, m)-space, which is symbolically denoted as $t/\tau\,\mathbf{P}$, and for parabola stacking $\mathbf{P}$ in (τ, m)-space it is weighted stacking $\tau/t\,\mathbf{H}$ in (t, x)-space. A dot product test was used to test these results (Table 0.1). The following form of the dot product test was used:

$$(\mathbf{L^T d}, \mathbf{L^T d}) = (\mathbf{L L^T d}, \mathbf{d}), \tag{13}$$

where $\mathbf{L}$ is $\mathbf{H^T}$, $\mathbf{P}$ or whatever operator was used.

From the table we can see that adjoint operators derived in the continuous domain are much less precise (relative error about 0.7-0.8%) than adjoint operators derived in the discrete domain (relative error 0.001%). Operators $\mathbf{H}$ and $\mathbf{P}$ in the last line of the table are not adjoint to each other, but they were tested to support the theoretical results from Appendix A (relative error about 2%).

SAMPLING IN VELOCITY DOMAIN

Velocity domain can be sampled evenly in velocity, velocity squared (alacrity), slowness or slowness squared (sloth). The aim of this section is to compare them from the point of view of inversion.

When equation (6) or equation (7) are being solved, the invertibility of the operator $\mathbf{HH^T}$ or $\mathbf{P^TP}$ is important. If we want to invert a velocity stack (equation (11)) or its modification (equation (12)), then we are interested in the invertibility of the operator $\mathbf{H^TH}$ or $\mathbf{PP^T}$.

Matrix expression of stacking operators

Stacking along hyperbolas is a linear operator mapping a two-dimensional (t, x)-space into two-dimensional (τ, m)-space. The operator may therefore be expressed as a matrix in a four-dimensional space. This is not convenient for our imagination. The representation of the operator in two dimensions is preferable. This can be achieved by ordering all the columns of (t, x)-space into one vector, and the same for (τ, m)-space. To get an idea of what the matrix looks like, let us write the equation transforming (t, x)-space into (τ, m)-space when the number of samples is six, two offsets and two sloths. The nearest neighbor interpolation was used.

$$
\begin{pmatrix}
\begin{pmatrix} u_{11} \\ u_{21} \\ u_{31} \\ u_{41} \\ u_{51} \\ u_{61} \end{pmatrix} \\
\begin{pmatrix} u_{12} \\ u_{22} \\ u_{32} \\ u_{42} \\ u_{52} \\ u_{62} \end{pmatrix}
\end{pmatrix}
=
\begin{pmatrix}
\begin{pmatrix}
. & 1 & . & . & . & . \\
. & 1 & . & . & . & . \\
. & . & 1 & . & . & . \\
. & . & . & 1 & . & . \\
. & . & . & . & 1 & . \\
. & . & . & . & . & 1
\end{pmatrix} &
\begin{pmatrix}
. & . & 1 & . & . & . \\
. & . & 1 & . & . & . \\
. & . & . & 1 & . & . \\
. & . & . & . & 1 & . \\
. & . & . & . & 1 & . \\
. & . & . & . & . & 1
\end{pmatrix} \\
\begin{pmatrix}
. & 1 & . & . & . & . \\
. & . & 1 & . & . & . \\
. & . & 1 & . & . & . \\
. & . & . & 1 & . & . \\
. & . & . & . & 1 & . \\
. & . & . & . & . & 1
\end{pmatrix} &
\begin{pmatrix}
. & . & . & 1 & . & . \\
. & . & . & 1 & . & . \\
. & . & . & 1 & . & . \\
. & . & . & . & 1 & . \\
. & . & . & . & . & 1 \\
. & . & . & . & . & .
\end{pmatrix}
\end{pmatrix}
\begin{pmatrix}
\begin{pmatrix} d_{11} \\ d_{21} \\ d_{31} \\ d_{41} \\ d_{51} \\ d_{61} \end{pmatrix} \\
\begin{pmatrix} d_{12} \\ d_{22} \\ d_{32} \\ d_{42} \\ d_{52} \\ d_{62} \end{pmatrix}
\end{pmatrix}
\tag{14}
$$

Each submatrix in the matrix represents **NMO** transformation (Claerbout, 1989). The transpose matrix applied on $\mathbf{u}$ gives $\tilde{\mathbf{d}}$:

$$
\begin{pmatrix}
\begin{pmatrix} \tilde{d}_{11} \\ \tilde{d}_{21} \\ \tilde{d}_{31} \\ \tilde{d}_{41} \\ \tilde{d}_{51} \\ \tilde{d}_{61} \end{pmatrix} \\
\begin{pmatrix} \tilde{d}_{12} \\ \tilde{d}_{22} \\ \tilde{d}_{32} \\ \tilde{d}_{42} \\ \tilde{d}_{52} \\ \tilde{d}_{62} \end{pmatrix}
\end{pmatrix}
=
\begin{pmatrix}
\begin{pmatrix}
. & . & . & . & . & . \\
1 & 1 & . & . & . & . \\
. & . & 1 & . & . & . \\
. & . & . & 1 & . & . \\
. & . & . & . & 1 & . \\
. & . & . & . & . & 1
\end{pmatrix} &
\begin{pmatrix}
. & . & . & . & . & . \\
1 & . & . & . & . & . \\
. & 1 & 1 & . & . & . \\
. & . & . & 1 & . & . \\
. & . & . & . & 1 & . \\
. & . & . & . & . & 1
\end{pmatrix} \\
\begin{pmatrix}
. & . & . & . & . & . \\
. & . & . & . & . & . \\
1 & 1 & . & . & . & . \\
. & . & 1 & . & . & . \\
. & . & . & 1 & 1 & . \\
. & . & . & . & . & 1
\end{pmatrix} &
\begin{pmatrix}
. & . & . & . & . & . \\
. & . & . & . & . & . \\
. & . & . & . & . & . \\
1 & 1 & 1 & . & . & . \\
. & . & . & 1 & . & . \\
. & . & . & . & 1 & .
\end{pmatrix}
\end{pmatrix}
\begin{pmatrix}
\begin{pmatrix} u_{11} \\ u_{21} \\ u_{31} \\ u_{41} \\ u_{51} \\ u_{61} \end{pmatrix} \\
\begin{pmatrix} u_{12} \\ u_{22} \\ u_{32} \\ u_{42} \\ u_{52} \\ u_{62} \end{pmatrix}
\end{pmatrix}
\tag{15}
$$

Generally, if we denote $\mathbf{d}_i$ as the i-th trace of a gather, $\mathbf{u}_i$ as the i-th trace of a velocity analysis panel and $\mathbf{N}_{ij}$ as an NMO matrix transforming i-th offset with j-th sloth, then the velocity stacking transformation may be expressed by the equation

$$
\begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_k \end{pmatrix}
=
\begin{pmatrix}
\mathbf{N}_{11} & \mathbf{N}_{12} & \dots & \mathbf{N}_{1n} \\
\mathbf{N}_{21} & \mathbf{N}_{22} & \dots & \mathbf{N}_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{N}_{k1} & \mathbf{N}_{k2} & \dots & \mathbf{N}_{kn}
\end{pmatrix}
\begin{pmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \vdots \\ \mathbf{d}_n \end{pmatrix}
\tag{16}
$$

If we want to compute $\mathbf{d}$ from $\mathbf{u}$ by least squares, then the matrix $\mathbf{N^T N}$

$$
\mathbf{N^T N} =
\begin{pmatrix}
\sum \mathbf{N}_{i1}^{\mathbf{T}} \mathbf{N}_{i1} & \sum \mathbf{N}_{i1}^{\mathbf{T}} \mathbf{N}_{i2} & \dots & \sum \mathbf{N}_{i1}^{\mathbf{T}} \mathbf{N}_{in} \\
\sum \mathbf{N}_{i2}^{\mathbf{T}} \mathbf{N}_{i1} & \sum \mathbf{N}_{i2}^{\mathbf{T}} \mathbf{N}_{i2} & \dots & \sum \mathbf{N}_{i2}^{\mathbf{T}} \mathbf{N}_{in} \\
\vdots & \vdots & \ddots & \vdots \\
\sum \mathbf{N}_{in}^{\mathbf{T}} \mathbf{N}_{i1} & \sum \mathbf{N}_{in}^{\mathbf{T}} \mathbf{N}_{i2} & \dots & \sum \mathbf{N}_{in}^{\mathbf{T}} \mathbf{N}_{in}
\end{pmatrix}
\tag{17}
$$

should be inverted. This matrix is not even approximately unitary, as it is for **NMO** (Claerbout, 1989). For our example we have

$$
\mathbf{N^T N} =
\begin{pmatrix}
. & . & . & . & . & . & . & . & . & . & . & . \\
. & 3 & . & . & . & . & . & . & 2 & 1 & . & . \\
. & . & 3 & . & . & . & . & . & . & 3 & . & . \\
. & . & . & 2 & . & . & . & . & . & . & 2 & . \\
. & . & . & . & 2 & . & . & . & . & . & 1 & 1 \\
. & . & . & . & . & 2 & . & . & . & . & . & 1 \\
. & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . \\
. & 2 & . & . & . & . & . & . & 2 & . & . & . \\
. & 1 & 3 & . & . & . & . & . & . & 4 & . & . \\
. & . & . & 2 & 1 & . & . & . & . & . & 3 & . \\
. & . & . & . & 1 & 1 & . & . & . & . & . & 2
\end{pmatrix}
\tag{18}
$$

Hence just a transpose operation to stacking cannot be used for inverting the velocity analysis panel.

The rank of the matrix $\mathbf{N^TN}$ cannot be higher than $max(n.nt, k.nt)$, where nt is the number of samples of a trace. From this it follows that the matrix $\mathbf{N^TN}$ is singular if the number of traces in the (τ, m)-space is less than the number of traces in (t, x)-space. Generally, this matrix may be expected to be singular even for the opposite case, as we can see from equation (18).

Examples of matrices

Some matrices were generated to compare the properties of inversion dependent on sampling in velocity space. When we want to obtain a velocity panel by solving equation (7), then the matrix $\mathbf{HH^T}$ should be inverted. When we want to get data back from the velocity stack panel, then the matrix $\mathbf{H^TH}$ should be inverted.

It would be best to take the parameters of some real CMP gather and generate the corresponding matrices. The amount of computer time required for such computations would be, however, too large. With the following parameters I was able to obtain the results in a reasonable time: $n1 = 20$, $o1 = 0$ sec, $d1 = 0.1$ sec, $n2 = 10$, $o2 = 0$ km, $d2 = 0.2$ km. Number of velocities in most computations was chosen to be 10. The value that differs most from the real values is the sampling interval in time. This is probably the reason why so many singular values are practically equal to zero (Figure **??**).

Let us begin with the operator $\mathbf{H^TH}$, i.e. the operator needed for inversion of velocity stack. The matrix $\mathbf{H}$ is in Figure **??**a. We can see clearly how it is composed of NMO operators. Using only one velocity $v = \infty$ km/sec in the velocity stack leads to the matrix in Figure **??**b. This is a singular matrix, which corresponds to the fact that the original data cannot be recovered from this stack. The velocity stack for ten velocities in the range $1.5 - 8.5$ km/sec with even sampling in velocity and alacrity domains leads to the matrices in Figure **??**c and Figure **??**d. The high limit of velocity $v = 8.5$ km/sec was chosen, because it is the maximum resolvable velocity in the profile from Southern California in 4 msec sampling. Figure **??**e and Figure **??**f represent matrices with even sampling in the slowness and sloth domains. We can see that for sampling in velocity and alacrity spaces, the elements of matrices are pushed closer to the diagonals, which makes them more similar to the singular matrix in Figure **??**b. From this point of view sampling in slowness or sloth domains seems to have better properties.

The same argument holds for matrices $\mathbf{HH^T}$ (needed if inversion is used to compute the velocity panel) (Figure **??**a,b,c,d). By comparing Figure **??**c and Figure **??**d we see that sloth blurs elements of the matrix around diagonals slightly more than slowness and thus should be preferred.

$\mathbf{HH^T}$ for one velocity $v = \infty$ km/sec is a unit matrix (Figure **??**e) and hence regular, which agrees with the fact that horizontal stack of a gather is uniquely determined. The matrix for two velocities is in Figure **??**f. In this matrix, the number of diagonals linearly increases with the number of velocities, in contrast to the matrix $\mathbf{H^T H}$, in which the number of diagonals depends on the number of offsets.

Singular values

To compare the various ways of sampling in velocity domain, we should choose some criterion. A condition number of a matrix appears to be useless here, as we can see from Figure **??**. All the matrices are singular. Figure **??**c gives us a clue to the means of comparison. Two curves are plotted in this figure. One represents singular values of the matrix $\mathbf{H^T H}$ for ten velocities (the smoother one), and the other, for two velocities. The matrix for two velocities is evidently singular and the data cannot be restored when this matrix is used. The curve of singular values of this matrix drops faster to the zero level. The distribution of singular values for the matrix of ten velocities is more uniform.

Let us analyze the curves in Figure **??**a from this point of view. There are four curves in the figure, which from the left to the right (upper parts of the curves) represent singular values of matrices computed for even sampling in sloth, slowness, velocity and alacrity domains. This is the succession according to the above criterion. The velocity and alacrity curves have evidently worse behavior. The difference between the sloth and slowness curves is much less. The difference between the sloth and slowness curves in Figure **??**d is almost nil. In this figure high velocities were not included. In the region of high velocities slowness has worse behavior than sloth.

Singular values for matrices $\mathbf{H}$ are in Figure **??**b. They should be the square roots of those in Figure **??**a. This relationship holds for about the first 150 singular values. It does not hold for the rest of the values because of the limited computational precision (double precision used).

Geometrical representation of $\mathbf{H^T H}$

In Appendix B an expression for the operator $\mathbf{H^T H}$ is derived. It says that the value of the operator $\mathbf{H^T H}$ at a point (t, x) is equal to a weighted sum of all values lying on the hyperbolas passing through the point (t, x). The hyperbolas correspond to the velocities used in velocity analysis. Operators for various kinds of sampling are shown in Figure **??**. From these figures we can judge which sampling to choose.

Each hyperbola represents an equation with up to nx variables, where nx is the number of offsets. Up to nv hyperbolas may pass through every point of the gather, where nv is the number of velocities. So each point appears in up to nv equations. If some hyperbolas are close to each other, it means that the corresponding equations are almost identical. From Figure **??** we can see that this phenomenon happens with sampling in velocity and alacrity domains. For the same number of velocities we may actually have fewer independent equations than when sampling is done in the sloth domain. For sampling in the slowness domain, it seems that velocity analysis should not be done up to infinite velocity (Figure **??**c). If some reasonable limit of velocity is chosen, then the operator is comparable to the operator with evenly sampled sloth. These results agree with the results obtained by use of singular value decomposition.

This representation of $\mathbf{H^T H}$ can be also used for an explanation of why far offsets converge more slowly than near offsets in the conjugate gradient method (Figure **??** and Figure **??**d).

COUNT

For NMO using nearest neighbor interpolation, count is used to get inverse NMO (Claerbout, 1985). Count forms a diagonal of the matrix $NMO^T NMO$. We can see the shape of this diagonal in Figure **??**a. Count corresponds to the jacobian in the continuous case. The expression for the jacobian is derived in Appendix C.

Claerbout (1987) introduces $count(x,s,t)$ for velocity analysis as a count of times that a point located on the mesh at (x,t) will be added into slowness s. For the purposes of this section $count(t,x)$ is introduced as a count of times that a point located on the mesh at (x,t) will be added into velocity stack, so that we can write

$$count(t,x) = \sum_s count(x,s,t). \tag{19}$$

For linear interpolation $count(t,x)$ may be defined as a sum of weights with which a point will be added into velocity stack. For NMO we will get a curve in Figure **??**b. In both cases a theoretical count (jacobian) is drawn through the curves. Count may be expressed with the operator $\mathbf{H}$ as (after some normalization)

$$count(t,x) = \mathbf{H^T H} d(t,x), \tag{20}$$

where

$$d(t,x) = 1 \tag{21}$$

for all t and x. This can be seen from the way adjoint operators are computed (spreading).

Count for a CMP gather is shown in Figure **??**. We can see that the line $t = x/v_{min}$ is the line of the maximum count. Count on each trace is equal to the sum of NMO counts of this trace for each velocity involved in velocity analysis (Appendix B and C). This is apparent from Figure **??**c.

Velocity analysis was done for fifty velocities. From Figure **??**c we can see that in most of the gather below the line $t = x/v_{vmin}$ count does not differ much from 50. This implies that $count(x,s,t)$ does not differ much from 1 in this part of the gather. This result corresponds to NMO, where the same result holds tru truee.

The area above the line $t = x/v_{max}$ is not covered by any hyperbola, so that this part of the gather cannot be restored. It contributes to the singularity of the operator $\mathbf{H^T H}$.

Choice of lower limit of velocity

When the operator $\mathbf{H}^{\mathbf{T}}\mathbf{H}$ is applied to a gather, then the area around the line $t = x/v_{min}$ is susceptible to amplification. From this, it follows that the low limit of velocity in velocity analysis should not be chosen to be too small, otherwise this effect may distort the data. The low limit should not be chosen too high, either, because then hyperbolas with lower velocities cannot be restored in a low number of iterations. Thus the most convenient choice appears to be the lowest velocity in a gather.

CONCLUSIONS

Velocity analysis using a conjugate gradient method gives a better resolution than classical stacking along hyperbolas. Furthermore it enables us to easily get the original data from the velocity analysis panel. Modifications of basic operators using weighting etc. may be useful if we want to use only one iteration.

A comparison of the various ways of sampling in the velocity domain shows that even sampling in slowness squared should be chosen. It would be worth trying this comparison for matrices of sizes corresponding to the real gathers, but the amount of computer time required for this task discouraged me from doing so.

The minimum velocity for velocity analysis should not be chosen too low, because the data that is obtained from the velocity analysis panel might be spoiled by noise.

ACKNOWLEDGEMENTS

I would like to thank Bill Harlan for suggesting this topic to me and providing me with his code. My thanks also belong to Jon F. Claerbout for attracting my attention to some of his papers, which helped me understand better some concepts and find their applications. His inquisitive questions at seminars directed my attention to problems that I otherwise would not have noticed. I want to express my gratitude to Gilles Darche for allowing me to use his processed version of BP Alaska gather and for many helpful discussions. I am grateful to Francis Muir for supervising my work in previous quarters, which helped me a lot in my further work.

REFERENCES

Claerbout, J.F.,1985, What is the transpose operation: SEP-**42**, 113-128.

Claerbout, J.F., 1987, The velocity transform and its preconditioning: SEP-**56**, 137-143.

Claerbout, J.F., 1989, Book3 or revised parts of FGDP: SEP-**60**, 303-347.

Jedlička, J., 1989a, Stochastic normal moveout: SEP-**60**, 165-178.

Jedlička, J.,1989b, A truncation effect in velocity analysis: SEP-**61**.

Thorson, J.R. and Claerbout, J.F., 1985, Velocity-stack and slant-stack stochastic inversion: Geophysics, **50**, 2727-2741

Thorson, J.R., 1984, Velocity and slant stack inversion: Ph.D. thesis, Stanford University.

APPENDIX A

ADJOINT OPERATORS TO STACKING OPERATORS

A natural extension of dot products in (t, x)-space and (τ, m)-space in a continuous domain is

$$(\mathbf{d}_1, \mathbf{d}_2) = \iint d_1(t,x) d_2(t,x)\, dt\, dx, \tag{22}$$

$$(\mathbf{u}_1, \mathbf{u}_2) = \iint u_1(\tau,m) u_2(\tau,m)\, d\tau\, dm. \tag{23}$$

The definition of an adjoint operator in a continuous domain cannot be done straightforwardly by transposing a corresponding matrix (we simply do not have the matrix). Instead another equivalent property is used for definition:

$$(\mathbf{Lu}, \mathbf{d}) = (\mathbf{u}, \mathbf{L^T d}). \tag{24}$$

This is the equation that is used for the dot product tests (Claerbout, 1985).

The definition of a velocity stacking operator $\mathbf{H}$ in (t, x)-space is

$$u(\tau, m) = \int d(\sqrt{\tau^2 + x^2 m}, x)\, dx. \tag{25}$$

Similarly we can define a stacking operator $\mathbf{P}$ in (τ, m)-space:

$$d(t, x) = \int u(\sqrt{t^2 - x^2 m}, m)\, dm. \tag{26}$$

Now the question is what is the relationship between the two operators. It can be shown (Thorson, 1984) that they are adjoint for dot products defined by equations

$$(\mathbf{d}_1, \mathbf{d}_2) = \iint t\, d_1(t,x) d_2(t,x)\, dt\, dx, \tag{27}$$

$$(\mathbf{u}_1, \mathbf{u}_2) = \iint \tau\, u_1(\tau,m) u_2(\tau,m)\, d\tau\, dm. \tag{28}$$

They are not, however, adjoint for the dot products defined by equations (23). If we want to apply an algorithm using Euclidean norms, then our operators should be adjoint with respect to the Euclidean norms.

Let us derive the adjoint operator $\mathbf{H^T}$:

$$(\mathbf{u}, \mathbf{Hd}) = \iint dm\, d\tau\, u(m,\tau) \int dx\, d(\sqrt{\tau^2 + x^2 m}, x) \tag{29}$$

$$= \iiint dm\, d\tau\, dx\, u(m,\tau) d(\sqrt{\tau^2 + x^2 m}, x) \tag{30}$$

After substitution $t = \sqrt{\tau^2 + x^2 m}$, $d\tau = t/\tau\, dt$ we obtain

$$(\mathbf{u}, \mathbf{Hd}) = \iint dt\, dx\, d(t,x) \int dm\, u(\sqrt{t^2 - x^2 m}, m) \frac{t}{\sqrt{t^2 - x^2 m}}. \tag{31}$$

Finally we have

$$\mathbf{H^T u} = \int u(\sqrt{t^2 - x^2 m}, m) \frac{t}{\sqrt{t^2 - x^2 m}}\, dm. \tag{32}$$

Similarly the adjoint operator $\mathbf{P^T}$ may be derived:

$$\mathbf{P^T d} = \int d(\sqrt{\tau^2 + x^2 m}, x) \frac{\tau}{\sqrt{\tau^2 + x^2 m}}\, dx. \tag{33}$$

For short reference we will write the above equations symbolically

$$\mathbf{H^T} = \frac{t}{\tau}\mathbf{P} \tag{34}$$

$$\mathbf{P^T} = \frac{\tau}{t}\mathbf{H} \tag{35}$$

We may weight both operators in the offset and velocity space. Here an infinite number of weighting factors can be obtained and a problem is which pair of them to choose. Here are some of them:

offset	1	τ/t	$\sqrt{\tau/t}$	$1/t$	τ
sloth	t/τ	1	$\sqrt{t/\tau}$	$1/\tau$	t

APPENDIX B

INTEGRAL EXPRESSION OF THE OPERATOR $\mathbf{H^T H}$

The aim of this section is to derive the expression for the operator $\mathbf{H^T H}$. We know that

$$(\mathbf{u}_1, \mathbf{Hd}_2) = (\mathbf{H^T u}_1, \mathbf{d}_2) = \iint \mathbf{d}_2(t,x) \left(\int \mathbf{u}_1(\sqrt{t^2 - x^2 m}, m) \frac{t}{\sqrt{t^2 - x^2 m}} dm\right) dt\, dx \tag{36}$$

for any $\mathbf{u}_1$ from (τ, m)-space and $\mathbf{d}_2$ from (t, x)-space. If we define

$$\mathbf{u}_1(\tau, m) = \int \mathbf{d}_1(\sqrt{\tau^2 + y^2 m}, y) dy, \tag{37}$$

then

$$(\mathbf{H^T H d}_1, \mathbf{d}_2) = \iint \mathbf{d}_2(t, x)(\iint \mathbf{d}(\sqrt{t^2 - (x^2 - y^2)m}, y) \frac{t}{\sqrt{t^2 - x^2 m}} dy) dm. \tag{38}$$

From this equation it follows

$$\mathbf{H^T H d}(t, x) = \iint \mathbf{d}(\sqrt{t^2 - (x^2 - y^2)m}, y) \frac{t}{\sqrt{t^2 - x^2 m}} dy\, dm, \tag{39}$$

which is the expression for the operator $\mathbf{H^T H}$.

Equation $T = \sqrt{t^2 - (x^2 - y^2)m}$ represents a hyperbola in (T, y)-space passing through the point (t, x). With changing m we obtain a set of hyperbolas passing through the point (t, x) (Fig **??**). A weighting factor is assigned to each hyperbola. The higher the slope of a hyperbola is, the higher weighting factor is assigned. The value of the operator $\mathbf{H^T H}$ at the point (t, x) is equal to the weighted sum of all values lying on the hyperbolas passing through the point (t, x).

This result may also be explained in the following way. The sums along hyperbolas passing through the point (t, x) form a parabola in (τ, m)-space (Jedlička, 1989). $\mathbf{H^T H}(t, x)$ is obtained by weighted summation along this parabola. Each point of the parabola represents a sum along a corresponding hyperbola, from which the previous result follows immediately.

If $d(t, x) = 1$ for all t and x then from equation (39) it follows

$$\mathbf{H^T H d}(t, x) = C \int \frac{t}{\sqrt{t^2 - x^2 m}} dm. \tag{40}$$

This means that $count(t, x)$ for velocity analysis is equal to the sum of NMO counts over the range of velocities used in velocity analysis. An expression for a count for NMO is derived in Appendix C.

APPENDIX C

JACOBIAN FOR NMO

Let us denote $\mathbf{N}$ the NMO operator and $\mathbf{N^T}$ its transpose operator. The matrix $\mathbf{N^T N}$ is diagonal for nearest neighbor interpolation (Claerbout, 1989). To see how the coefficients of the diagonal of $\mathbf{N^T N}$ look like, let us compute the operator $\mathbf{N^T N}$

in the continuous domain. For any traces $\mathbf{d}_1$ and $\mathbf{d}_2$ it holds

$$\begin{aligned}
(\mathbf{Nd}_1, \mathbf{Nd}_2) &= \int \mathbf{Nd}_1(\tau)\mathbf{Nd}_2(\tau) && (41)\\
&= \int \mathbf{d}_1(\sqrt{\tau^2 + x^2 m})\mathbf{d}_2(\sqrt{\tau^2 + x^2 m})d\tau && (42)\\
&= \int \mathbf{d}_1(t)\mathbf{d}_2(t)\frac{t}{\sqrt{t^2 - x^2 m}}dt && (43)\\
&= (\mathbf{d}_1, \mathbf{N^T N d}_2) && (44)
\end{aligned}$$

so that we have

$$\mathbf{N^T N d}_2(t) = \frac{t}{\sqrt{t^2 - x^2 m}}\mathbf{d}_2(t). \qquad (45)$$