Stanford Exploration Project, Report 75, November 17, 1997, pages 235–263

Stanford Exploration Project, Report 75, November 17, 1997, pages 235–263

Green's function traveltime interpolation

David E. Lumley[1]

keywords: *traveltime, velocity, interpolation, modeling, ray tracing*

ABSTRACT

Green's function traveltime fields are useful for Kirchhoff depth migration and inversion applications, and for tomographic velocity inversion. However, in 2-D, and especially 3-D, traveltimes become increasingly computationally expensive. Based on the physical continuity of (first arrival) traveltimes, I derive a least squares expression from the ray theoretic eikonal equation, which allows for the interpolation of known Green's function traveltime functions from several surface source positions, to one or more Green's function traveltime functions at intermediate arbitrary source positions. The least squares equation is parameterized in terms of the traveltime *gradient*, and is solved by standard weighted least-squares techniques and Singular Value Decomposition, and then integrated up to traveltime directly. For a 2-D (3-D) geometry, at least two (three) traveltime fields must be known in order to interpolate the unknown source traveltime field. I test the theory and algorithm on a constant velocity model, constant velocity gradient model, and the complex 2-D structural velocity model of Marmousi. The results are very encouraging to date, although some sparse numerical instability in the calculation of the traveltime gradient fields must be overcome in order to be of practical use.

INTRODUCTION

Traveltimes from a given source position to a regular grid of subsurface positions can be effectively calculated by finite difference solution of the ray theoretic eikonal equation (Vidale, 1988, Van Trier and Symes, 1991), or by raytracing techniques such as the paraxial method (Beydoun and Keho, 1987). Since the traveltime field is associated with the phase of the WKBJ Green's function, it can be useful in integral solution applications such as Kirchhoff prestack depth migration (Keho and Beydoun, 1988, and Van Trier, 1991) and tomographic inversion (Luo and Schuster, 1990).

In such applications, the traveltime field, or its Frechet derivative, is required at each subsurface location, due to effective sources at each receiver and shot posi-

[1]**email:** not available

tion. In 2-D, this rapidly becomes a computational burden, and in 3-D a seemingly insurmountable obstacle. Usually, in areas of significant lateral migration velocity variation, traveltime fields are required at every source and receiver position. For surface seismic data, this requirement ideally means at every surface CMP location along a line, at a 10-30 m spacing. For a depth migration along a 2-D 10 km line, this means evaluating several hundred traveltime tables, each of which may consist of 10^5–10^6 mesh points.

In a "slow" CPU environment, the traveltime fields may be precalculated prior to migration or inversion, stored on disk, and then accessed remotely during the migration or inversion procedure. Or, in a *very fast* CPU environment, each traveltime field may be regenerated redundantly "on the fly" as needed. The precompute strategy relieves some of the computational effort, but can be extremely taxing in terms of available disk storage and I/O bandwidth bottlenecks. The "on the fly" strategy pushes the limits of present-day supercomputing capabilities for realistically large 2-D prestack depth migration problems, and is probably out of reach for 3-D prestack depth migration. As an example, Cabrera et al. (1992) presented one of the few state-of-the-art 3-D prestack depth migrations publicly discussed, and used a massively parallel supercomputer (Intel iPSC/860) to precompute traveltimes at every 20 CMPs in a quasi $v(z)$ background velocity model. The 3-D prestack depth migration was done as a second step, by retrieving the necessary traveltime values from the lookup tables stored on disk.

An alternate strategy to the two outlined above, would be to precompute a *very sparse* amount of traveltime fields over the survey area, thus reducing both computational effort and disk storage, and then subsequently interpolating the desired traveltimes from the given sparse set. I have in mind a goal of something on the order of 500 m spacings, even in complex laterally variable media. By reason of physical continuity in the (first arrival) traveltime field, there must be some mathematical basis for extrapolating and/or interpolating such traveltime fields from adjacent source positions.

With this in mind, I derive both an extrapolation equation and a system of interpolation equations for the traveltime field, based on the ray theoretic eikonal. In the first section of this paper, I briefly review the eikonal equation. Next, I derive an extrapolation equation to find the traveltime field due to a source, given the traveltime field of an adjacent source position. This is then extended to a system of interpolation equations, given several adjacent traveltime fields. I discuss how the interpolation system can be solved by standard least-squares or SVD techniques. Finally, I test the theory and the ensuing algorithm on velocity models that are: constant, constant gradient, and complex 2-D (Marmousi). The interpolation is found to be highly accurate for smooth (but strong) velocity gradient fields, and somewhat less accurate for the complex Marmousi velocity model. However, the results are encouraging and this work is still only in its infancy.

THEORY

Traveltime extrapolation

Suppose I know the Green's function traveltime field $\tau_1(\mathbf{x};\mathbf{x}_1)$ for all subsurface coordinates $\mathbf{x}$ due to a source at a (surface) position $\mathbf{x}_1$. Estimating the traveltime field $\tau_2(\mathbf{x};\mathbf{x}_2)$ due to an adjacent surface source position $\mathbf{x}_2$, from the given field τ_1, is defined as traveltime *extrapolation*.

I assume that τ_1 has been previously calculated by solving the ray-theoretic eikonal equation (Červený et al., 1977):

$$|\nabla \tau_1(\mathbf{x};\mathbf{x}_1)|^2 = s^2(\mathbf{x}) \tag{1}$$

where ∇ is the vector gradient operator, and $s(\mathbf{x})$ is the earth slowness (inverse velocity) field. The unknown field τ_2 can be decomposed without loss of generality as follows:

$$\tau_2(\mathbf{x};\mathbf{x}_2) = \tau_1(\mathbf{x};\mathbf{x}_1) + \phi(\mathbf{x};\mathbf{x}_1,\mathbf{x}_2) \tag{2}$$

where ϕ is the traveltime difference between the known field τ_1 and the unknown field τ_2, and is a function of subsurface location $\mathbf{x}$ and the two specified source positions $\mathbf{x}_1$ and $\mathbf{x}_2$. Naturally, one requires that the extrapolated field τ_2 also satisfies the eikonal:

$$|\nabla \tau_2|^2 = |\nabla \tau_1 + \nabla \phi|^2 = s^2(\mathbf{x}) \ . \tag{3}$$

Expanding the squared term in (3) and substituting (1) results in an equation for ϕ in terms of the known field τ_1:

$$2\nabla \tau_1 \cdot \nabla \phi + |\nabla \phi|^2 = 0 \ . \tag{4}$$

In this form, solving for ϕ looks harder than solving for τ_2 directly from the eikonal equation! However, given that $\phi = \tau_2 - \tau_1$, the term $|\nabla \phi|^2$ can be expanded as follows:

$$|\nabla \phi|^2 = 2\left(s^2 - \nabla \tau_1 \cdot \nabla \tau_2\right) = 2s^2\left(1 - \cos\theta_{12}\right) \tag{5}$$

where $\theta_{12}(\mathbf{x};\mathbf{x}_1,\mathbf{x}_2)$ is the angle at any subsurface point $\mathbf{x}$ between the ray connecting $\mathbf{x}_1$ to $\mathbf{x}$, and the ray connecting $\mathbf{x}_2$ to $\mathbf{x}$. Substituting (5) into (4), the equation for ϕ in terms of τ_1 becomes:

$$\nabla \tau_1 \cdot \nabla \phi + s^2\left(1 - \cos\theta_{12}\right) = 0 \ , \tag{6}$$

which can also be expressed as an equation for τ_2 directly in terms of τ_1:

$$\nabla\tau_1 \cdot \nabla\tau_2 - s^2 \cos\theta_{12} = 0 \; . \tag{7}$$

This last equation is simply a statement that the dot product of the two gradient fields is the cosine of the angle between the local ray directions, scaled by the local value of the slowness squared.

If $\cos\theta_{12}$ can be approximated in some physically reasonable and robust manner, then (6) or (7) represent a single first-order linear PDE to extrapolate the unknown field τ_2. I believe such an approximation is attainable, since only the angle between the two rays is required, and not each ray angle individually. In fact, if the surface source location $\mathbf{x}_2$ is on the order of a few tens of meters away from $\mathbf{x}_1$, and we are interested in extrapolating traveltimes at imaging points a few kilometers distant from the source region, then $\cos\theta$ can not stray too far from a value of unity, based on arguments of traveltime field continuity. This is a mathematical reinforcement of our intuition that, if we perturb our source location a bit, then the resulting perturbed traveltime field should not be tremendously different from the unperturbed field. Many of us have noted this effect while tracing traveltime maps along a line for prestack depth migration, and have been frustrated that we were forced to completely recalculate *slightly* different traveltime fields from CMP to adjacent CMP.

As a first approximation, the function $\cos\theta_{12}$ could be approximated by the Law of Cosines, as:

$$\cos\theta_{12} \;\sim\; \frac{r_1^2 + r_2^2 - |\mathbf{x}_2 - \mathbf{x}_1|^2}{2r_1 r_2} \tag{8}$$

where r_1 is the (straight) ray distance from $\mathbf{x}_1$ to $\mathbf{x}$, and r_2 is the (straight) ray distance from $\mathbf{x}_2$ to $\mathbf{x}$. Although this constant velocity approximation would seem to be inappropriate for estimating ray angles in general $s(\mathbf{x})$ media, it may not be so bad for estimating the angle *between* the two rays, as discussed above. In this sense, the constant velocity assumption means more like: the velocity field "seen" along ray 1 is about the same as the velocity field "seen" along ray 2. Note that this approximation is expected to deteriorate as the source separation $|\mathbf{x}_2 - \mathbf{x}_1|^2$ increases or the ray lengths r_1, r_2 decrease. Conversely, the approximation (8) is expected to be good for cases in which $|\mathbf{x}_2 - \mathbf{x}_1| << \min\{r_1, r_2\}$.

Both (6) and (7) can be cast into the general 3-D form:

$$a\psi_x + b\psi_y + c\psi_z + d = 0 \; , \tag{9}$$

or the 2-D form:

$$a\psi_x + c\psi_z + d = 0 \; , \tag{10}$$

where $\{a, b, c\}$ are the components of the gradient vector of the known field $\{\tau_{1x}, \tau_{1y}, \tau_{1z}\}$, d is a function of squared slowness s^2 and $\cos\theta_{12}$, and $\{\psi_x, \psi_y, \psi_z\}$ are the components of the unknown traveltime gradient field, $\{\tau_{2x}, \tau_{2y}, \tau_{2z}\}$ or $\{\phi_x, \phi_y, \phi_z\}$, which are to be solved for. The extrapolation equation (9) can be solved by a finite difference method, for example, and represents a first-order linear PDE which should require less effort to solve than the nonlinear eikonal PDE for τ_2. The solution will be in terms of the *gradient* of the unknown field, $\nabla\tau_2$, which can then be integrated to the traveltimes $\tau_2(\mathbf{x})$ directly (refer to Equation (18)). This completes the extrapolation process of an unknown traveltime field $\tau_2(\mathbf{x};\mathbf{x}_2)$ from a known traveltime field $\tau_1(\mathbf{x};\mathbf{x}_1)$. This extrapolation equation might be useful for computing Frechet derivatives of the type

$$\frac{\partial\tau(\mathbf{x};\mathbf{x}_s)}{\partial\mathbf{x}_s} \sim \phi(\mathbf{x};\mathbf{x}_s+\delta\mathbf{x}_s) \tag{11}$$

for applications in data acquisition survey design, or shot gather seismic data continuation, for example.

Traveltime interpolation

Now suppose I know a set of n Green's function traveltime fields $\tau_i(\mathbf{x};\mathbf{x}_i)$ for n source positions $\mathbf{x}_i$. Estimating the traveltime field $\tau_k(\mathbf{x};\mathbf{x}_k)$, due to a single intermediate surface source position $\mathbf{x}_k$, from the n given fields τ_i, is defined as traveltime *interpolation.*

For the set of τ_i, (7) can be generalized to a linear matrix equation of the form:

$$\nabla\tau_i\cdot\nabla\tau_k - s^2\cos\theta_{ik} = 0\ ;\ i = 1, n\ , \tag{12}$$

or into a general weighted least-squares system $\mathcal{W}\mathcal{A}\mathbf{x} = \mathcal{W}\mathbf{y}$:

$$\begin{pmatrix} w_1 & & & & \\ & w_2 & & & \\ & & w_3 & & \\ & & & \ddots & \\ & & & & w_n \end{pmatrix} \begin{pmatrix} \tau_{1x} & \tau_{1y} & \tau_{1z} \\ \tau_{2x} & \tau_{2y} & \tau_{2z} \\ \tau_{3x} & \tau_{3y} & \tau_{3z} \\ \vdots & \vdots & \vdots \\ \tau_{nx} & \tau_{ny} & \tau_{nz} \end{pmatrix} \begin{pmatrix} \tau_{kx} \\ \tau_{ky} \\ \tau_{kz} \end{pmatrix} = s^2(\mathbf{x}) \begin{pmatrix} w_1\cos\theta_{1k} \\ w_2\cos\theta_{2k} \\ w_3\cos\theta_{3k} \\ \vdots \\ w_n\cos\theta_{nk} \end{pmatrix} . \tag{13}$$

System (13) can be solved by standard damped least-squares,

$$\mathbf{x} \sim (\mathcal{A}^T\mathcal{W}^T\mathcal{W}\mathcal{A} + \lambda\mathcal{I})^{-1}\mathcal{A}^T\mathcal{W}^T\mathcal{W}\mathbf{y}\ , \tag{14}$$

or by the optimal (but slow) Singular Value Decomposition (SVD) method, for the components of the unknown vector $\nabla\tau_k$. I refer the reader to Strang (1980) for an excellent review of damped least squares and SVD linear algebra techniques. The diagonal weighting matrix $\mathcal{W}$ is of dimensions (nxn), and the weights w_i could be inverse-distance: $w_i \sim |\mathbf{x}_i - \mathbf{x}_k|^{-1}$, for example. The $\mathcal{A}$ matrix is (nx3) in 3-D and (nx2) in 2-D (ignoring the ∂_y terms), and contains the vector components of the known traveltime gradient fields. The data vector $\mathbf{y}$ is (nx1) and contains the $\cos\theta_{ik}$ values. For a 2-D geometry, at least two traveltime gradient fields must be known, and for a 3-D interpolation at least three traveltime gradient fields must be available.

In the case of interpolation, a refinement to the estimate of the $\cos\theta_{ik}$ values is possible. In this paper, I use the Law of Cosines given by (8) as an initial estimate of, say, $\cos\theta_{12}(\mathbf{x})$ and $\cos\theta_{23}(\mathbf{x})$, where the traveltime field at $\mathbf{x}_2$ is to be interpolated from two bounding source locations $\mathbf{x}_1$ and $\mathbf{x}_3$. Please refer to Figure **??** to consider the appropriate interpolation geometry. However, the angle θ_{13} is known precisely from the given traveltime (gradient) fields at $\mathbf{x}_1$ and $\mathbf{x}_3$:

$$\cos\theta_{13} \equiv v^2 \nabla\tau_1 \cdot \nabla\tau_3 \ . \tag{15}$$

Therefore, given the angle estimates θ'_{12}, θ'_{23} and θ'_{13}, and the *true* angle θ_{13}, I can refine my initial (primed) estimates such that the total angle θ_{13} is conserved as a sum of θ_{12}, θ_{23}:

$$\theta_{13} = f\theta'_{13} = f(\theta'_{12} + \theta'_{23}) = \theta_{12} + \theta_{23} \ . \tag{16}$$

Then, the correction factor f is given as

$$f = \theta_{13}/\theta'_{13} \ ,$$

and

$$\theta_{12} \sim f\theta'_{12} \quad ; \quad \theta_{23} \sim f\theta'_{23} \ . \tag{17}$$

As an example, for a 2-D interpolation of τ_2 given τ_1 and τ_3, I first set up a 2x2 matrix system per (13). I evaluate the required data vector values $\cos\theta_{12}$ and $\cos\theta_{23}$ using (8), and refine the cosine estimates using (17). I then invert system (13) by Cramer's Rule if the determinant is not too small, or by SVD if otherwise. The entire procedure is repeated in a point-by-point independent manner, making it an obvious candidate for massively parallel implementation, for all subsurface grid locations $\mathbf{x}$.

Finally, the traveltime gradients are spatially integrated to traveltime such that, for a source at (x_2, z_2) on a flat surface $z = z_2$,

$$\tau_2(x,z) = \gamma(x, z_2) + \int_{z_2}^{z} \tau_z(x, z')\, dz' \ , \tag{18}$$

where

$$\gamma(x, z_2) = H(x - x_2) \int_{x_2}^{x} \tau_x(x', z_1)\, dx' - H(x_2 - x) \int_{x}^{x_2} \tau_x(x', z_1)\, dx' \ , \qquad (19)$$

where H is the Heaviside function. This process completes the traveltime interpolation of τ_2 from τ_1 and τ_3.

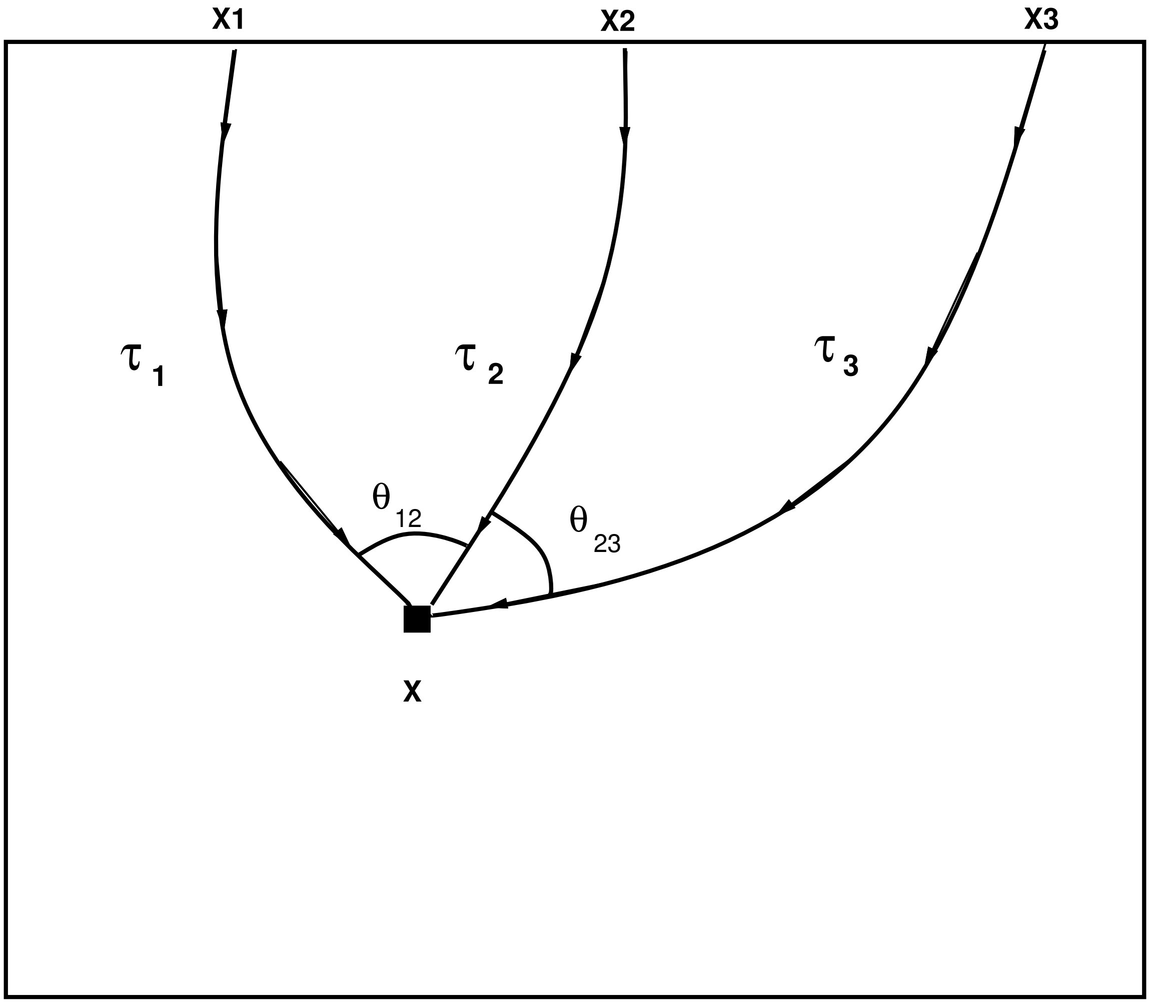


Figure 1: The traveltime interpolation geometry. It is assumed that the traveltime field $\tau(\mathbf{x}; \mathbf{x}_2)$ due to a source position at $\mathbf{x}_2$ can be interpolated from two known traveltime fields $\tau(\mathbf{x}; \mathbf{x}_1)$ and $\tau(\mathbf{x}; \mathbf{x}_3)$ due to sources at positions $\mathbf{x}_1$ and $\mathbf{x}_3$ respectively. david3-geometry [NR]

EXAMPLES

I now present three interpolation examples. The first example is a constant velocity medium, in order to get a feeling for the maximum potential accuracy and stability of the method. The second example is a strong constant gradient model in which the vertical and horizontal velocity gradients are both 1.0 km/s/km, and gives some intuition on the effects of lateral velocity variation. The third example is the "make or break" Marmousi model, to test the algorithm's response to a complex, highly contrasted 2-D velocity model.

Constant velocity

The first example is a constant velocity model of 2.0 km/s and dimensions of 6 km on a side. The first known source position is at $\mathbf{x}_1 = 0.0$ km, and the second known source position is at $\mathbf{x}_3 = 6.0$ km, which represents a very sparse traveltime table spatial separation of 6 km. (Of course, for constant velocity and $v(z)$ models, the traveltimes depend only on offset and depth, and are independent of lateral source position. In this regard, the comment on sparse sampling is relevant only in terms of interpolation, and is not relevant to what I would actually do in practice!) Their respective traveltime fields τ_1 and τ_3 are plotted in Figure **??**. The horizontal and vertical traveltime gradient fields at $\mathbf{x}_1$ are denoted as τ_{1x} and τ_{1z} respectively, and are plotted in Figure **??**. The horizontal and vertical traveltime gradient fields at $\mathbf{x}_2$ are denoted as τ_{3x} and τ_{3z} respectively, and are plotted in Figure **??**.

The interpolated traveltime field $\tau_2(\mathbf{x};\mathbf{x}_2)$ at $\mathbf{x}_2 = 3.0$ km is plotted in the top panel of Figure **??**, and the traveltime gradients τ_{2x} and τ_{2z} are plotted in Figure **??**. The relative interpolation traveltime error is contoured in the lower panel of Figure **??**. The contour farthest from the desired source position at 3.0 km has a value of 1%, and increases to 10% relative error as you move in toward the source position, in 1% contour value increments. In theory, this example should be error-free since the $\cos\theta$ terms are given *exactly* by (8) in a constant velocity medium. For this example, the interpolated traveltimes are accurate to within 1% relative error, my rule of thumb requirement, over most of the subsurface space, using a sparse shot interval of 6 km. However, there is evidence of some small numerical inaccuracy and a resulting tendency of singularity right at the source position, where the θ values change rapidly and are therefore most sensitive to error. Hence, in more general velocity models, we should expect instability near the source region.

Constant velocity gradient

The velocity model I now consider has a constant gradient, with both the vertical and horizontal velocity gradients equal to 1.0 km/s/km. The dimensions are the same as the previous constant velocity model. In this case, the velocity ranges linearly from

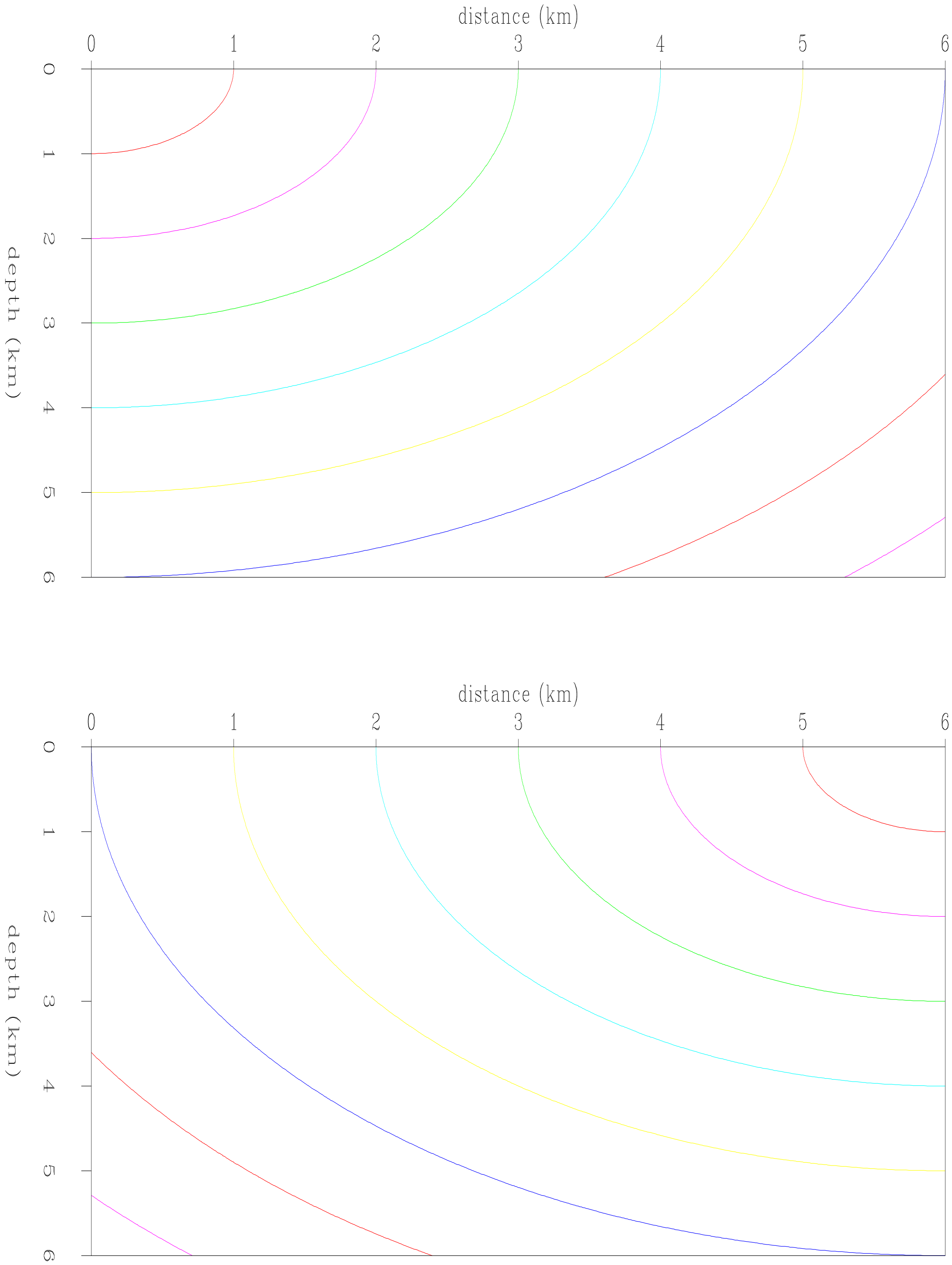


Figure 2: Constant velocity model. Contours of traveltime field for the two given surface source positions at 0.0 km (top) and 6.0 km (bottom). $\boxed{\text{david3-c13t}}$ [ER]

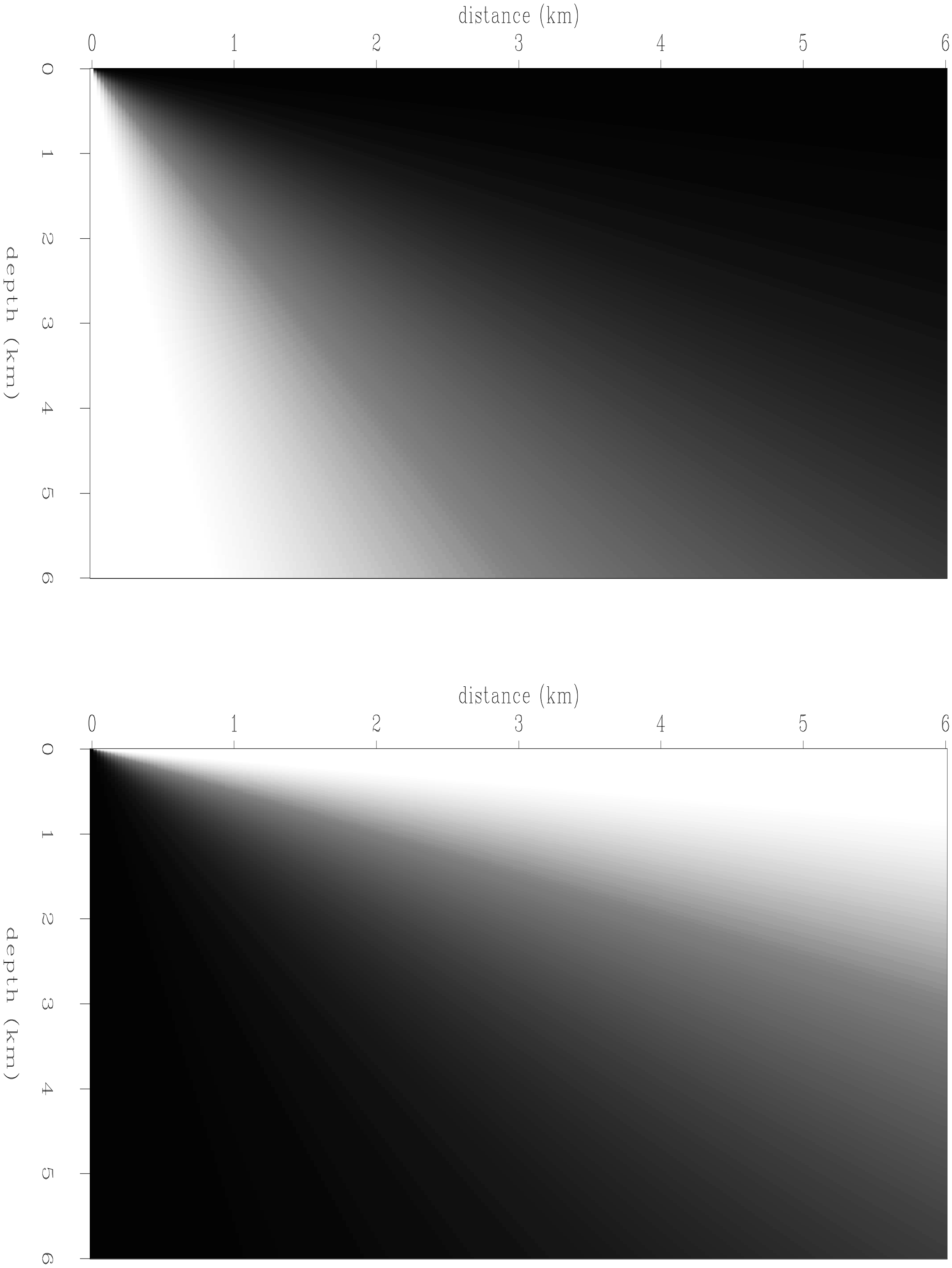


Figure 3: Constant velocity model. Horizontal (top) and vertical (bottom) traveltime gradient fields for the surface source positioned at 0.0 km. david3-c1xz [ER]

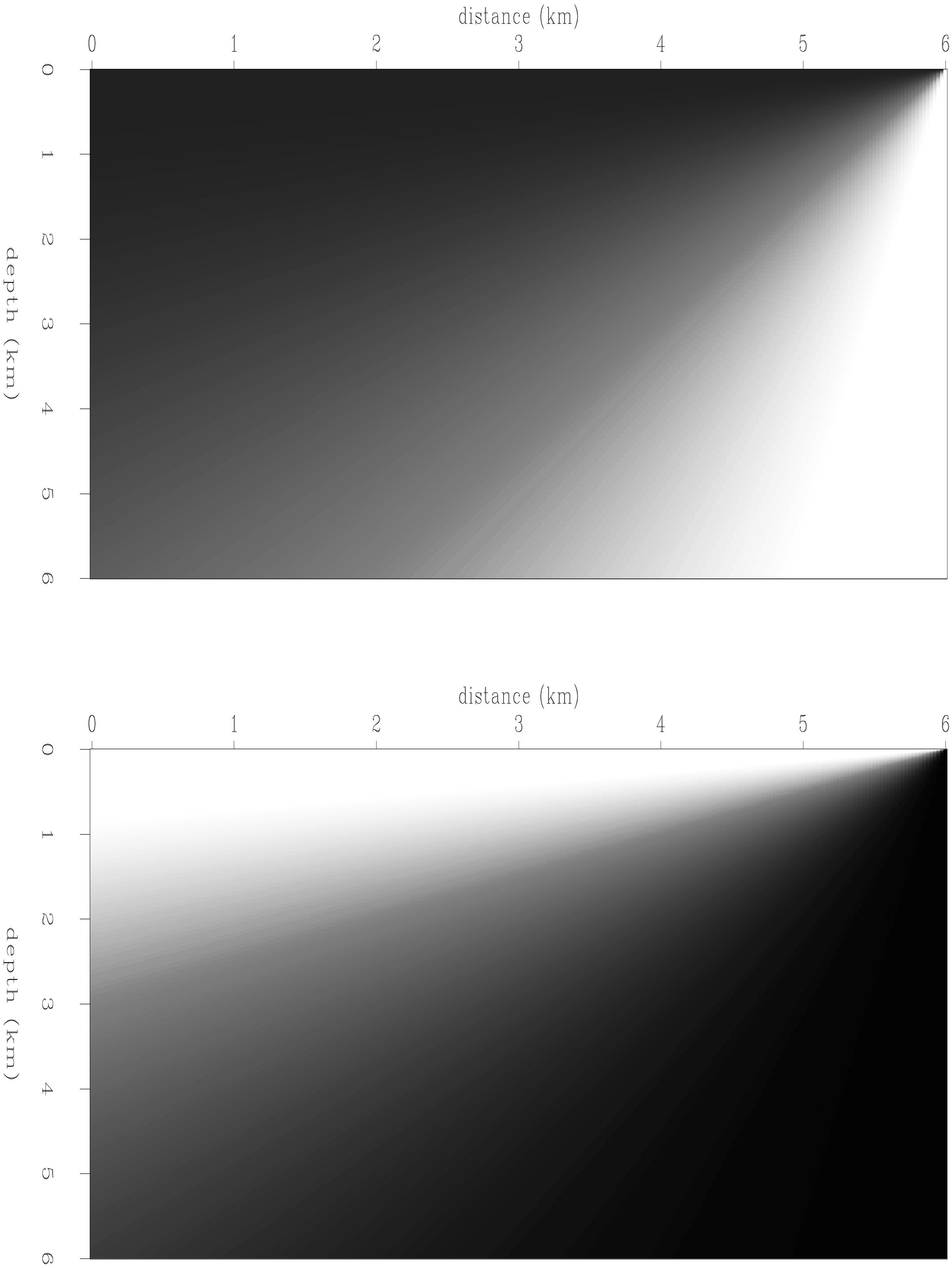


Figure 4: Constant velocity model. Horizontal (top) and vertical (bottom) traveltime gradient fields for the surface source positioned at 6.0 km. david3-c3xz [ER]

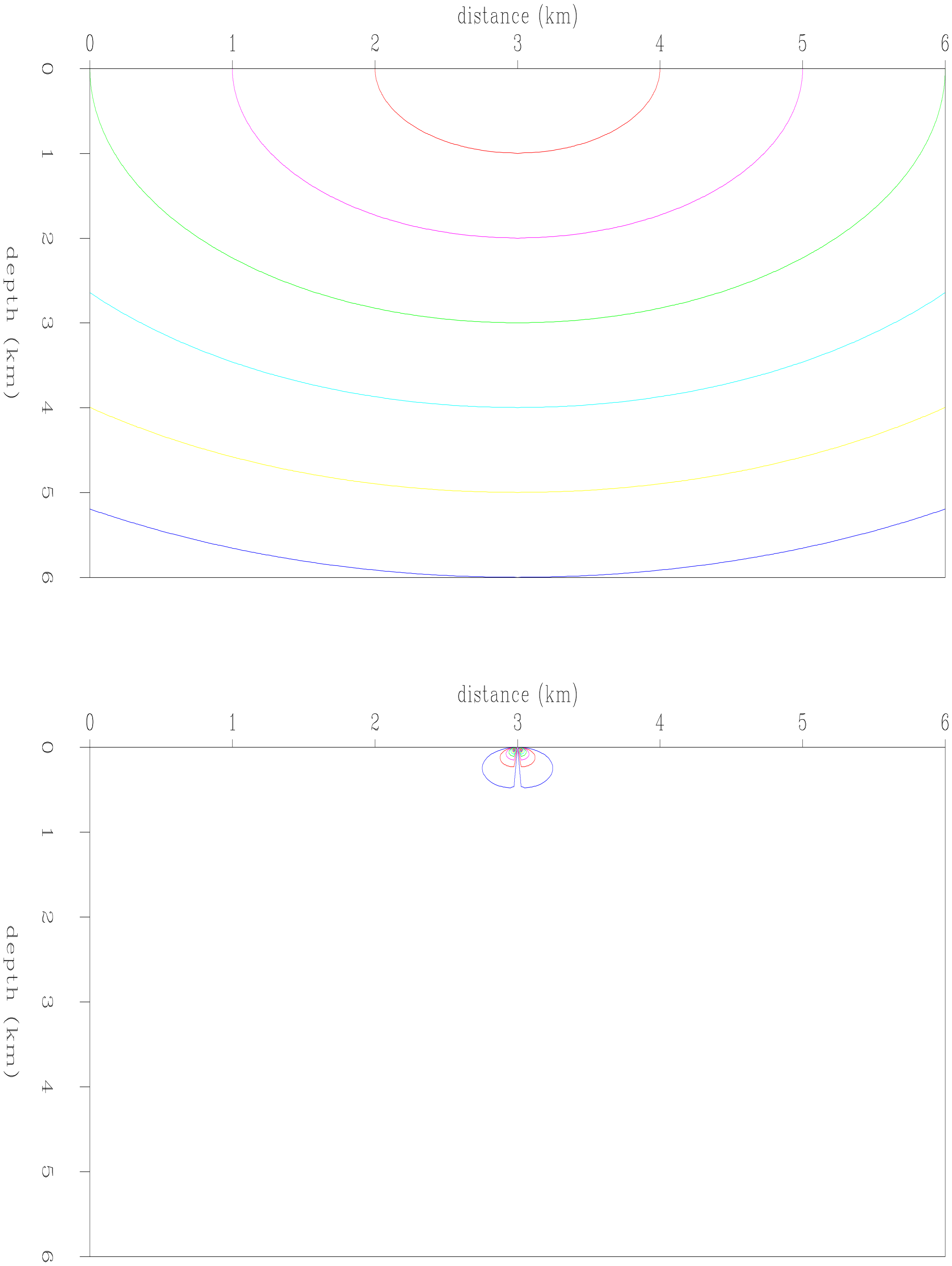


Figure 5: Constant velocity model. Interpolated traveltime field. The top panel is the interpolated traveltime field for a surface source positioned at 3.0 km. The lower panel is the relative error between the interpolated traveltime field, and the correct traveltimes modeled by an analytical solution to the eikonal. The contour farthest from the source region is at 1% relative error, and the contour values increase to 10% error right at the source location, in 1% contour increments. david3-cite [ER]

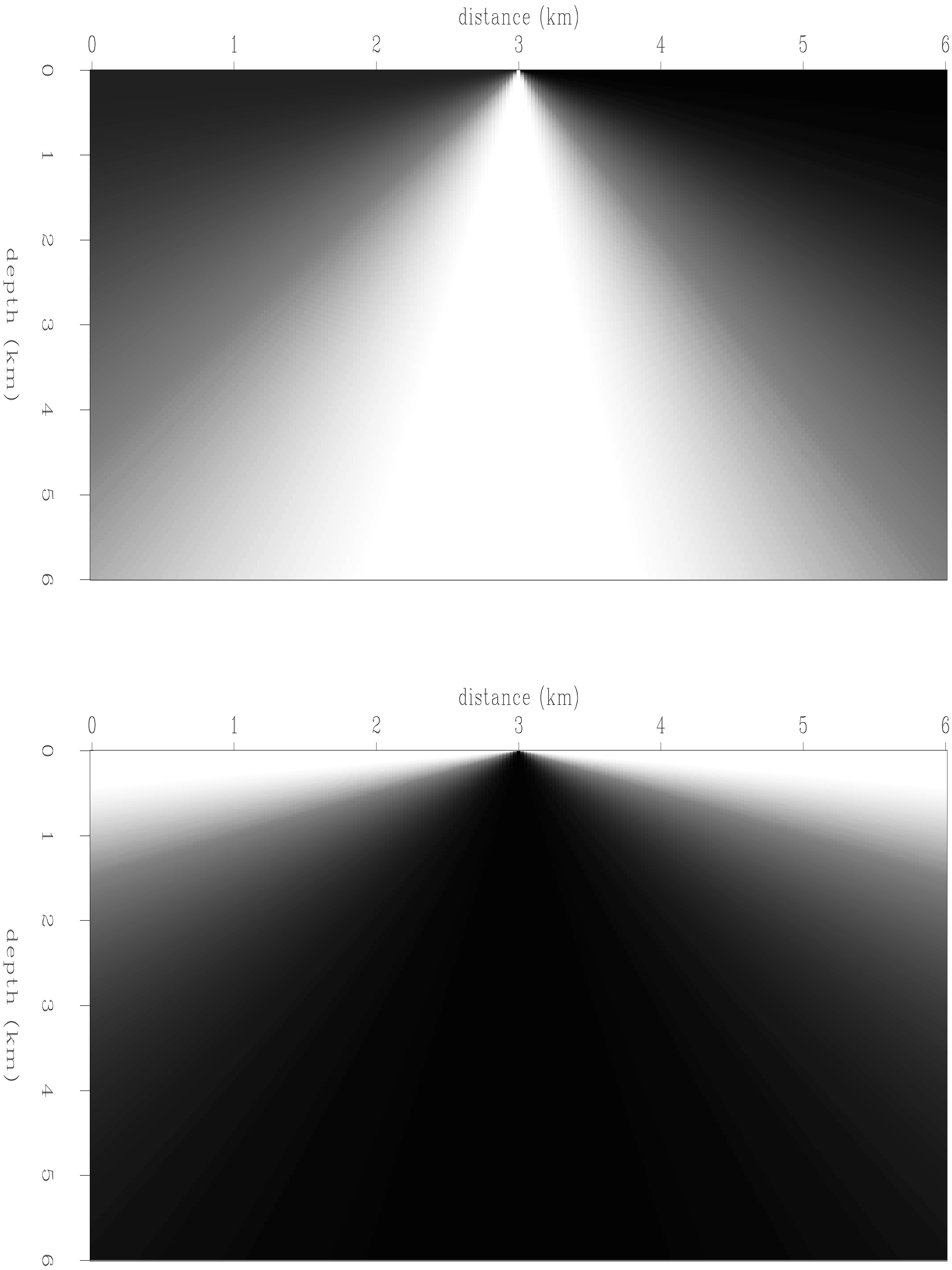


Figure 6: Constant velocity model. Horizontal (top) and vertical (bottom) traveltime gradient fields for the interpolated surface source positioned at 3.00 km. david3-cixz [ER]

1.0 km/s in the upper left corner to 13 km/s in the lower right corner, which is an extremely strong gradient compared to what may be naturally observed.

I presume that we have two previously calculated traveltime fields at surface source positions of $x = 2.75$ km and 3.25 km, i.e., a sparse traveltime table sampling every 500 m along the line (my other rule of thumb goal in addition to beating 1% relative error). Figure **??** shows the contours of traveltime due to the source at 2.75 km (upper panel), and due to a source at 3.25 km (lower panel). Since I interpolate traveltime gradients, and not the traveltimes themselves, Figures **??** shows the horizontal and vertical traveltime gradients τ_{1x} and τ_{1z} for the first source position, and Figure **??** shows the τ_{3x} and τ_{3z} for the source given at 3.25 km.

I then solve system (13) using a combination of Cramer's Rule where nonsingular, and Singular Value Decomposition otherwise. I also use the cosine estimate and correction discussed previously from (8) and (17). The interpolated traveltime field is shown in the upper panel of Figure **??** for a desired surface source position at 3.0 km. The lower panel of Figure **??** shows the relative error between the interpolated traveltime field, and the true traveltime field obtained by an analytic raytracing algorithm for constant gradient velocity media (Zhang, 1992, pers. comm.). The error contours are at 1% farthest from the source region, and increase to 10% in the near vicinity of the source. The interpolation is highly accurate, except for some small error directly beneath the source position. This is caused by the previously mentioned instability at the source position in the vertical traveltime gradient, and it is propagated somewhat into all of the traveltimes by the subsequent spatial integration. To help stabilize the singularity, I calculate the $\nabla\tau$ values within a few mesh points of the source location by assuming straight rays (constant velocity) from the source. Figure **??** shows the interpolated horizontal and vertical traveltime gradients respectively.

Marmousi velocity model

In this example, I try to break my code on the Marmousi velocity model, pictured in Figure **??**. The top panel is the true velocity model, and the lower panel is a 2-D smoothed version (150 m wavelength) I used for forward modeling traveltimes. Some smoothing is required to satisfy the ray validity conditions of the eikonal equation (Červený et al. (1977)), which most finite difference traveltime algorithms neglect. Velocity smoothing, ray validity conditions, and stable numerical evaluation of the eikonal equation as a function of classes of velocity models, are topics of utmost importance in traveltime calculation which are not presently well understood.

I used the hybrid finite difference local raytracing algorithm of Zhang (1992) to forward model traveltimes. Figure **??** show the traveltime field for a source at $\mathbf{x}_1 = 4.75$ km (upper panel), and a source at $\mathbf{x}_3 = 5.25$ km (lower panel). The contours show some areas, particularly in the lower panel near $(x, z) = (3.0,\ 2.2)$ km, where there are significant traveltime changes from $\mathbf{x}_1$ to $\mathbf{x}_3$ – a shot separation of 500 m. Due to this large change in traveltime behavior, and the cusp-like nature of the

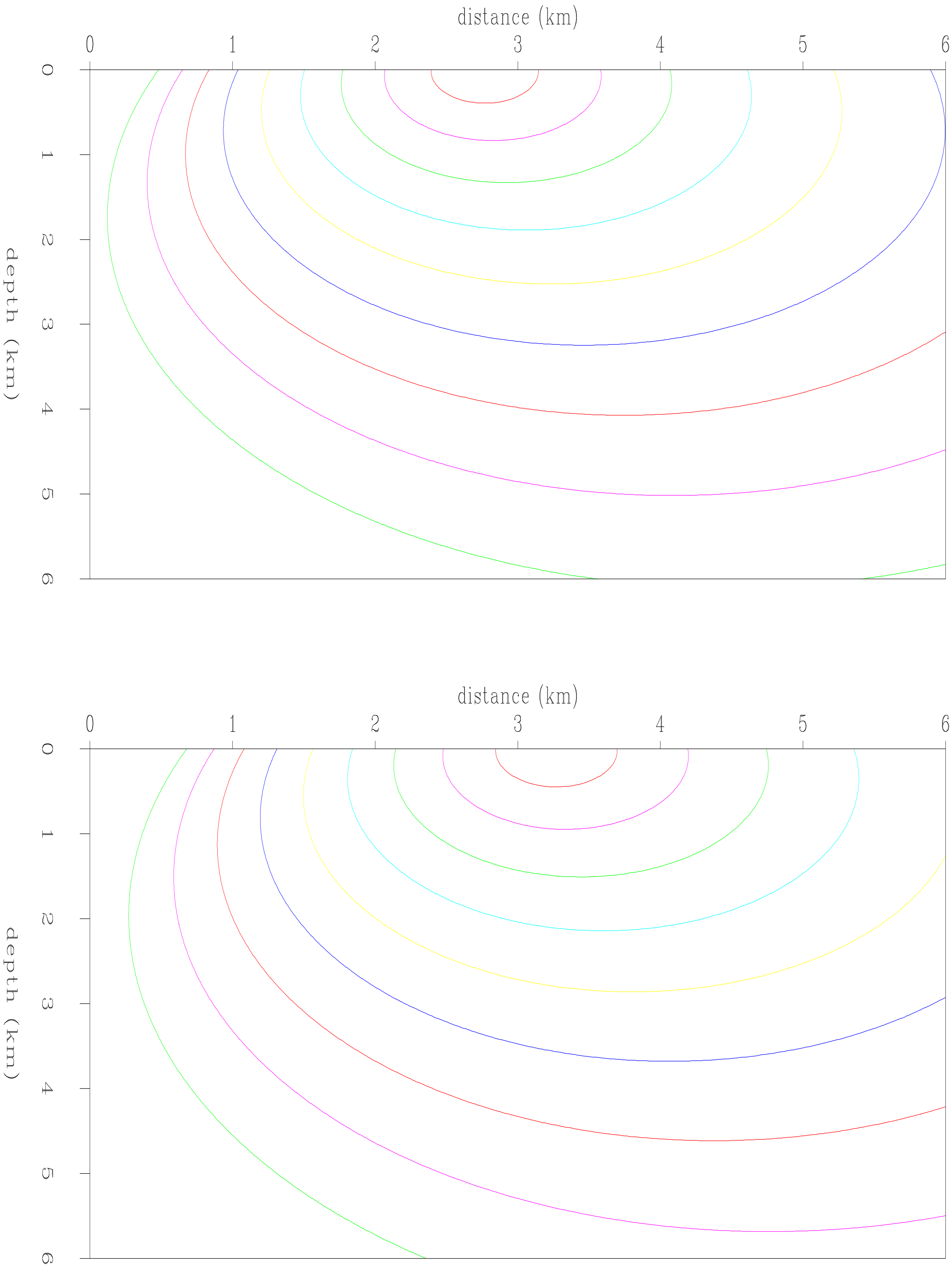


Figure 7: Linear gradient velocity model. Contours of traveltime field for the two given surface source positions at 2.75 km (top) and 3.25 km (bottom). david3-g13t [ER]

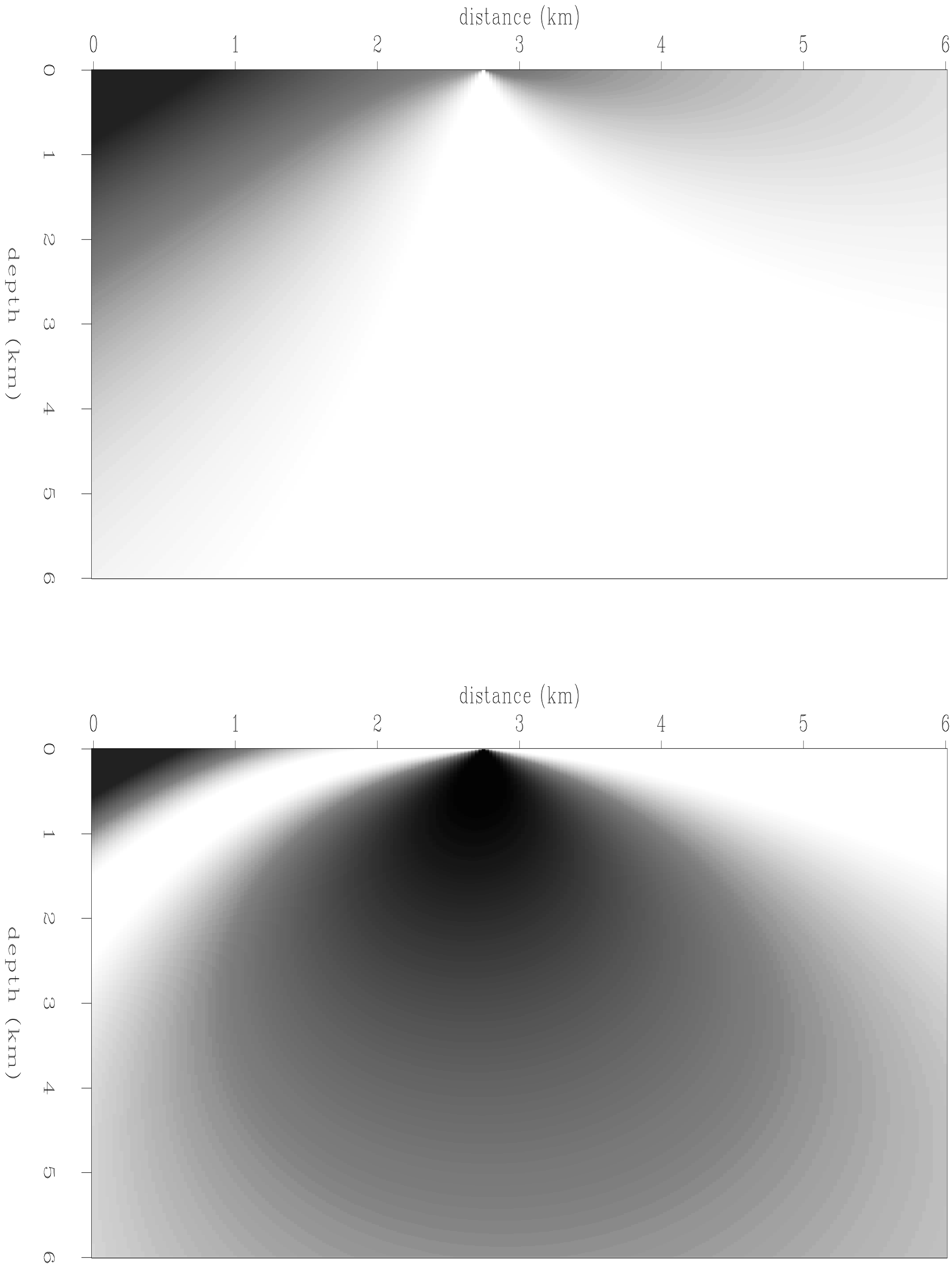


Figure 8: Linear gradient velocity model. Horizontal (top) and vertical (bottom) traveltime gradient fields for the surface source positioned at 2.75 km. david3-g1xz [ER]

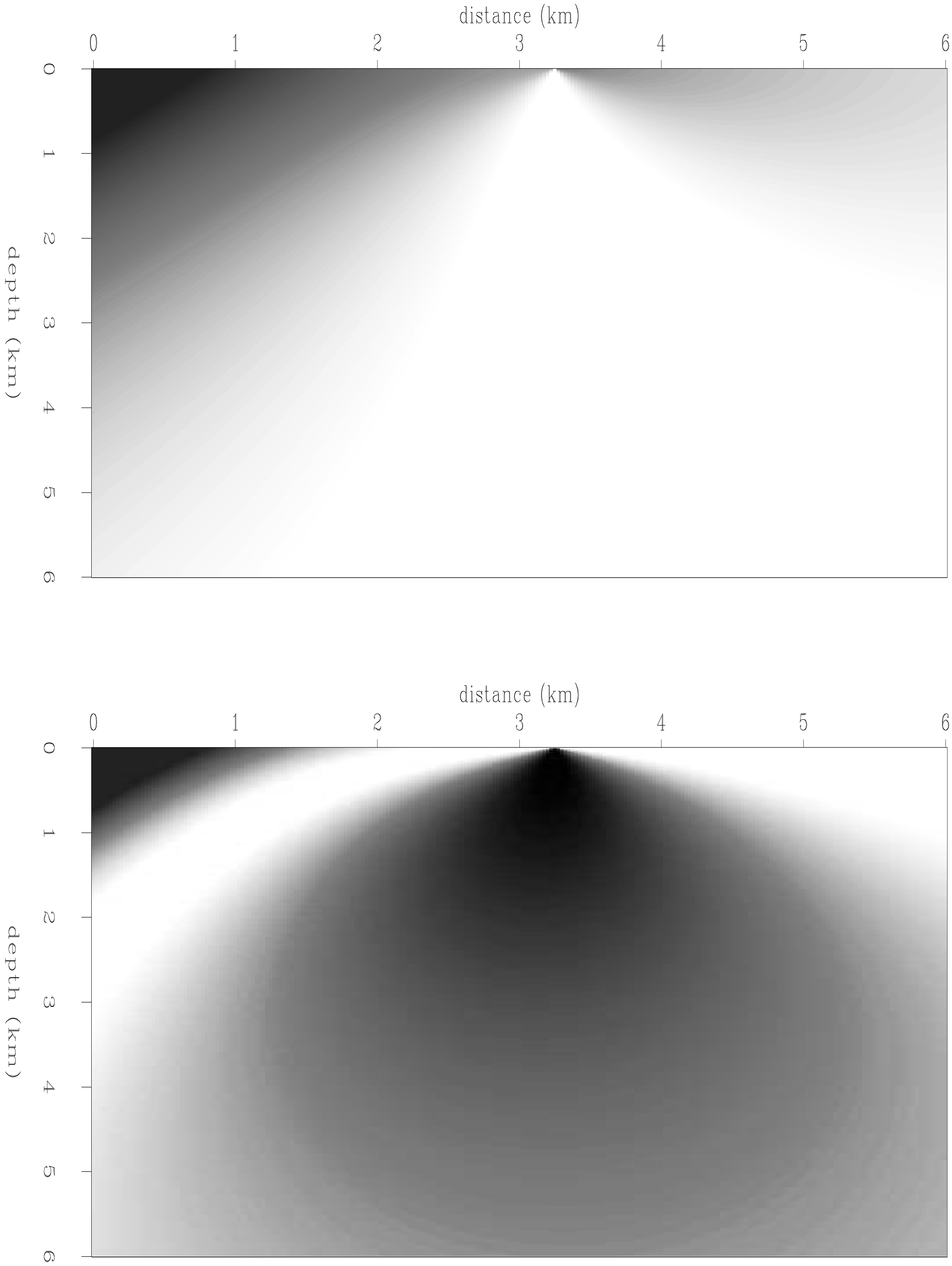


Figure 9: Linear gradient velocity model. Horizontal (top) and vertical (bottom) traveltime gradient fields for the surface source positioned at 3.25 km. david3-g3xz [ER]

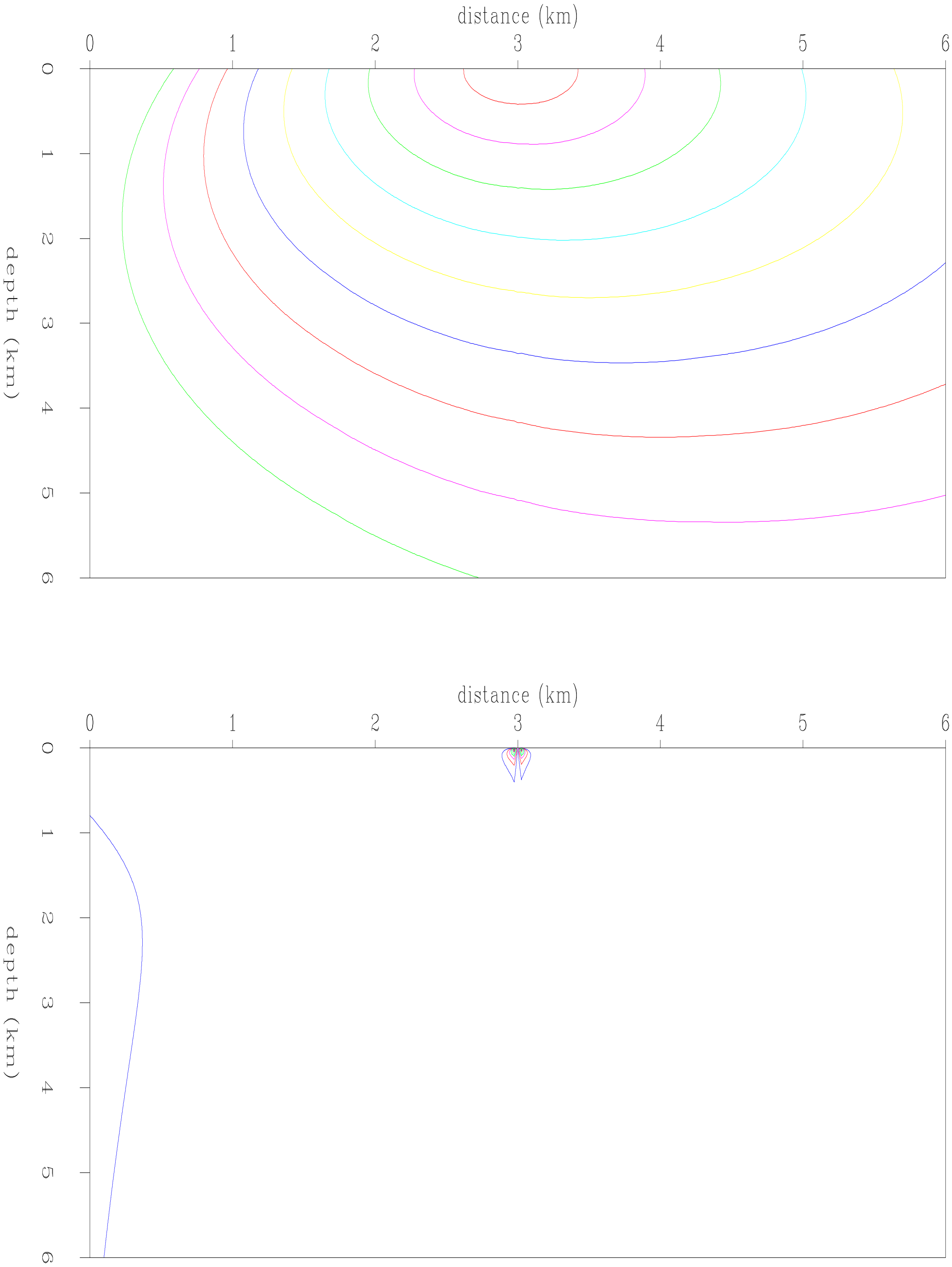


Figure 10: Linear gradient velocity model. Interpolated traveltime field. The top panel is the interpolated traveltime field for a surface source positioned at 3.00 km. The lower panel is the relative error between the interpolated traveltime field, and the correct traveltimes modeled by an analytical solution to the eikonal. The contour farthest from the source region is at 1% relative error, and the contour values increase to 10% error right at the source location, in 1% contour increments. david3-gite [ER]

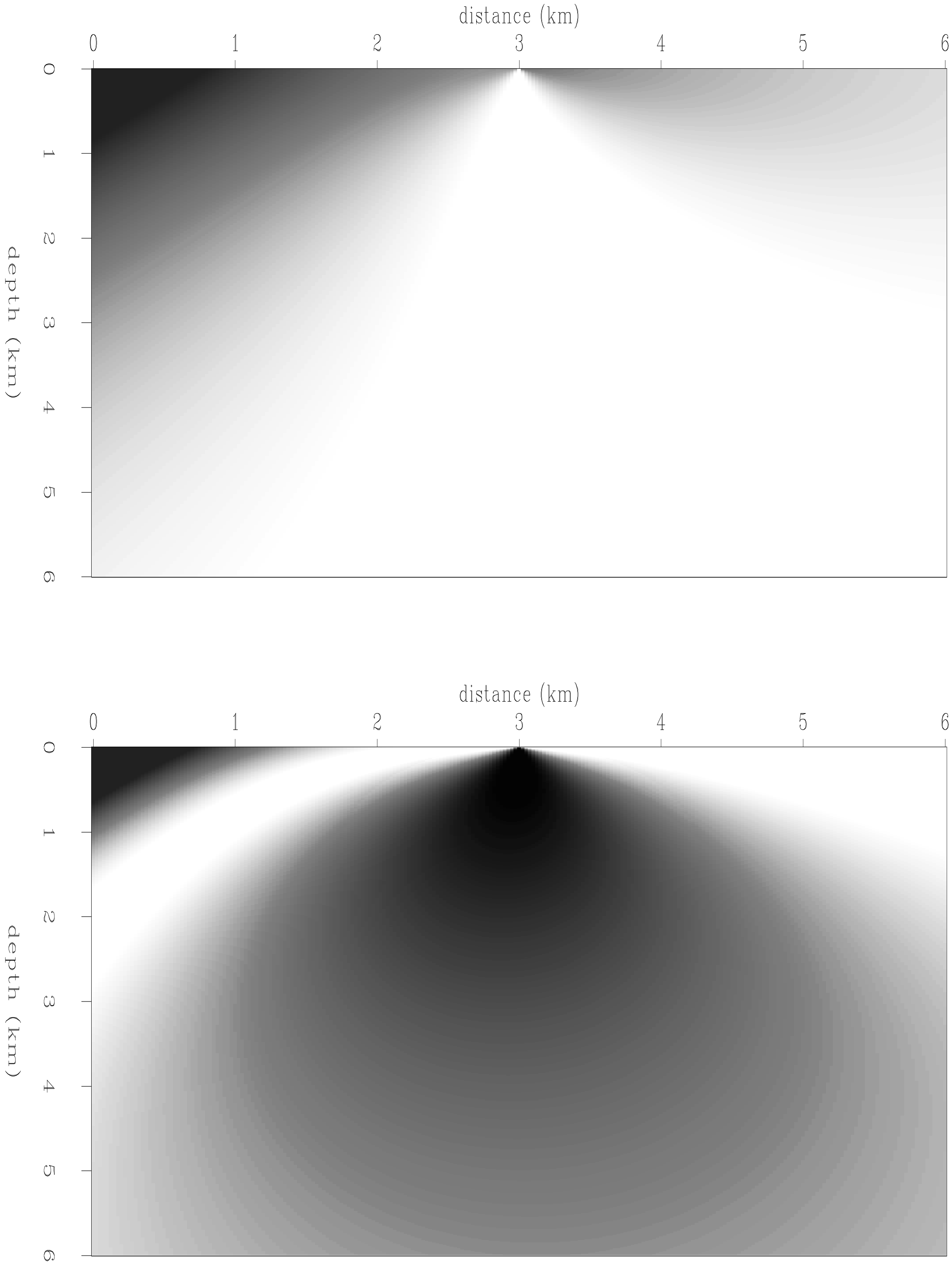


Figure 11: Linear gradient velocity model. Horizontal (top) and vertical (bottom) traveltime gradient fields for the interpolated surface source positioned at 3.00 km. david3-gixz [ER]

crossing traveltime branches in the lower panel, we can imagine that these areas will be difficult to interpolate, and they are. Figure **??** shows the traveltime gradients at source $\mathbf{x}_1$, and Figure **??** shows the traveltime gradients at source $\mathbf{x}_3$. These $\nabla\tau$ fields are amazingly discontinuous and predict trouble ahead.

Figure **??** shows the interpolated traveltime field for a source at 5.0 km. The lower panel shows the relative traveltime error in 5% contour intervals. The good news is that most of the subsurface region (uncontoured) has relative traveltime errors of less than 1%, at a fraction of the cost and disk storage compared to the modeling code. The bad news is that the corner regions at far offset and large depths are contaminated with traveltime errors in the range of 5–20%. Figure **??** explains it all. At wide offsets, with sharp velocity contrasts still remaining in the smoothed Marmousi model, complicated interactions between head waves and precritical refractions are causing large traveltime gradient contrasts. In this region, even my enhanced $\cos\theta$ estimate is breaking down, causing the evidenced traveltime instability. As the traveltimes are spatially integrated through these bad zones, the error is smeared out to larger depths and offsets.

However, before I sound the death knell, let's review. The algorithm did a fine interpolation for a large part of the Marmousi model. The interpolation was done at a 500 m shot spacing over a very complex velocity structure. Presently the interpolation code runs at about 4 times the speed of the traveltime forward modeling code, and I have made zero effort to optimize it. There is some traveltime gradient interpolation instability in regions of large and rapid velocity contrast that remain to be understood and overcome. These instabilities caused relative errors of 5–20% in the worst regions.

Future research

There are several things I would like to try in the future on the Marmousi example. Perhaps I didn't smooth the velocity model enough? How is the smoothing and the ability to interpolate related to the ray validity conditions? Perhaps a 500 m source spacing was too much to ask for, perhaps a 250 m would have be fine? Perhaps solving an overdetermined least squares system (my examples were exactly determined systems) might help to compensate for traveltime gradient irregularity? Is there an intelligent method to smooth or (median?) filter the traveltime gradient fields (input and interpolated) before spatial integration? Finally, is there a better approach which interpolates for the traveltime directly, instead of first finding traveltime gradients and then integrating? If there were, then that would minimize the penalty of having a few interpolation errors being smeared throughout large portions of the final interpolated traveltime result.

CONCLUSION

Starting from the ray-theoretic eikonal equation, I derived an equation to extrapolate source traveltimes, and a system of least squares equations to interpolate Green's function traveltime fields. The equations are cast in terms of traveltime gradient, and must be subsequently integrated to traveltime. I solved the least squares system by a combination of Cramer's Rule and SVD. The interpolation results were very impressive for a strong constant gradient velocity model, with relative traveltime errors generally much less than 1%. The results for the complex 2-D Marmousi velocity model were less impressive, with unstable interpolation results in localized regions of 5–20% relative traveltime error. These results suggest that this interpolation method might be very good for velocity models with strong vertical and lateral velocity gradients, provided those gradients are *smooth*. In this case, I expect the interpolation to be at least an order of magnitude faster in CPU time than direct forward modeling of the eikonal equation, and require at least an order of magnitude less disk or memory storage of precomputed traveltime tables. For velocity models with strong and rapid spatial contrasts, the interpolation may cause unacceptable traveltime errors due to the inherent violation of the eikonal ray validity conditions.

I am currently working to make this algorithm robust and practical. If successful, this method could greatly reduce disk/memory storage and CPU requirements for traveltime intensive applications such as 3-D prestack depth migration and 3-D traveltime tomography.

ACKNOWLEDGMENTS

I would like to thank Lin Zhang very much for our discussions together regarding finite difference Green's function calculations, and for graciously lending me his code to forward model the constant gradient and Marmousi initial traveltime fields required as input to my interpolation algorithm. Thanks, Lin!

REFERENCES

Cabrera, J., J., Perkins, W. T., Ratcliff, D. W., and Lynn, W., 1992, 3-D prestack depth migration on a massively parallel computer: implementation and case history: 54th Mtg. Eur. Assoc. Expl. Geophys., Abstracts, 268–269.

Červený, V., Molotkov, I. A., and Pšenčík, I., 1977, Ray method in seismology: Univerzita Karlova, Praha.

Beydoun, W. B., and Keho, T. H., 1987, The paraxial ray method: Geophysics, **52**, 1639–1653.

Keho, T. H., and Beydoun, W. B., 1988, Paraxial ray Kirchhoff migration: Geophysics, **53**, 1540–1546.

Luo, Y., and Schuster, G. T., 1990, Wave-equation traveltime inversion: 60th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1207–1210.

Strang, G., 1980, Linear algebra and its applications: Academic Press.

Van Trier, J., 1991, A massively parallel implementation of prestack Kirchhoff depth migration: 53rd Mtg. Eur. Assoc. Expl. Geophys., Abstracts.

Van Trier, J., and Symes, W. W., 1991, Upwind finite-difference calculation of traveltimes: Geophysics, **56**, 812–821.

Vidale, J. E., 1988, Finite-difference calculation of traveltimes: Bull. Seis. Soc. Am., **78**, 2062–2076.

Zhang, L., 1992, Wavefront propagation by local ray-tracing: this report.

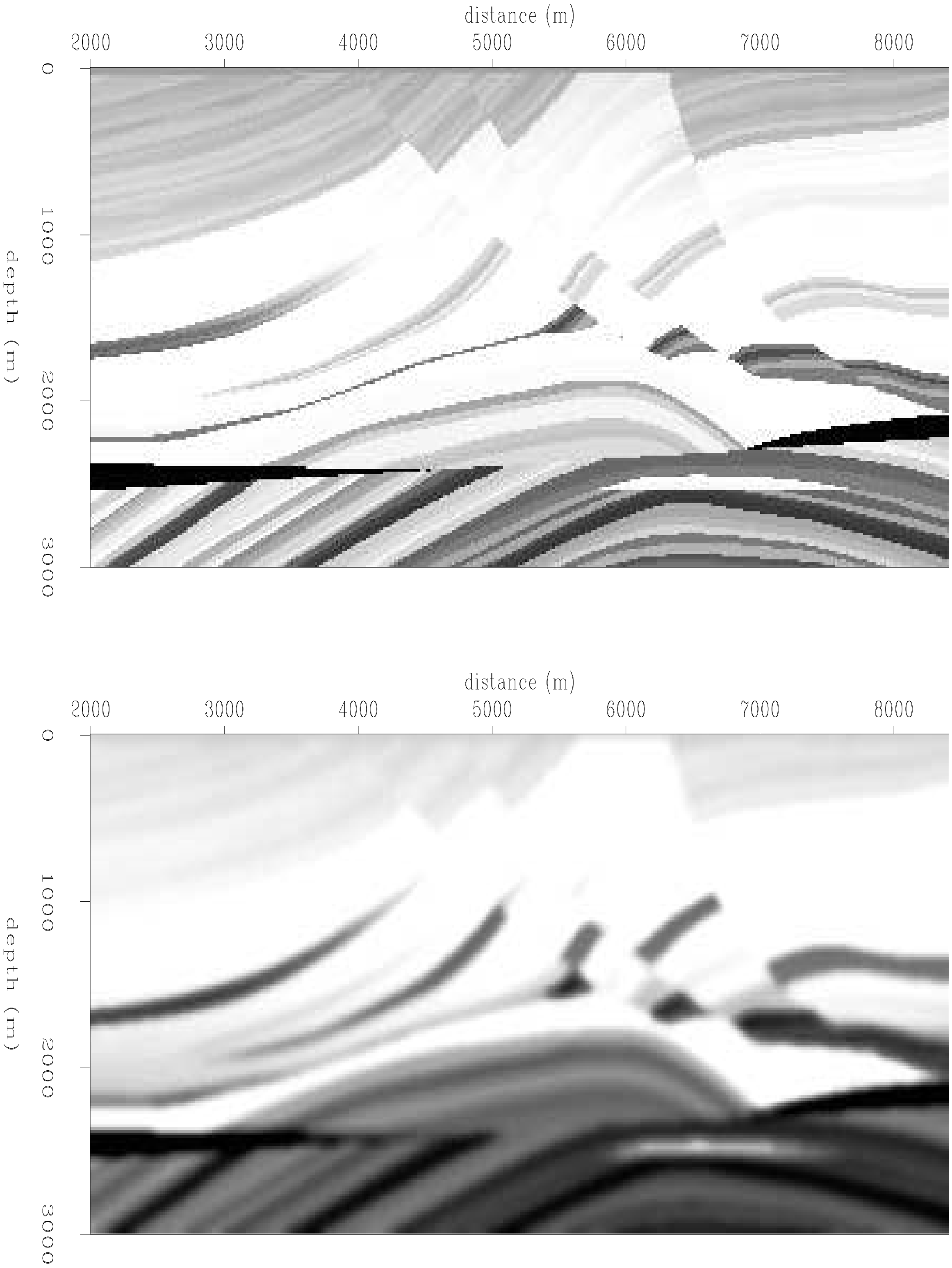


Figure 12: Marmousi 2-D velocity model. Top panel is original velocity model, lower panel is a smoothed version using a pyramid of base half-width equal to 80 m, which is used for the traveltime and interpolation calculations. david3-marmousi [ER]

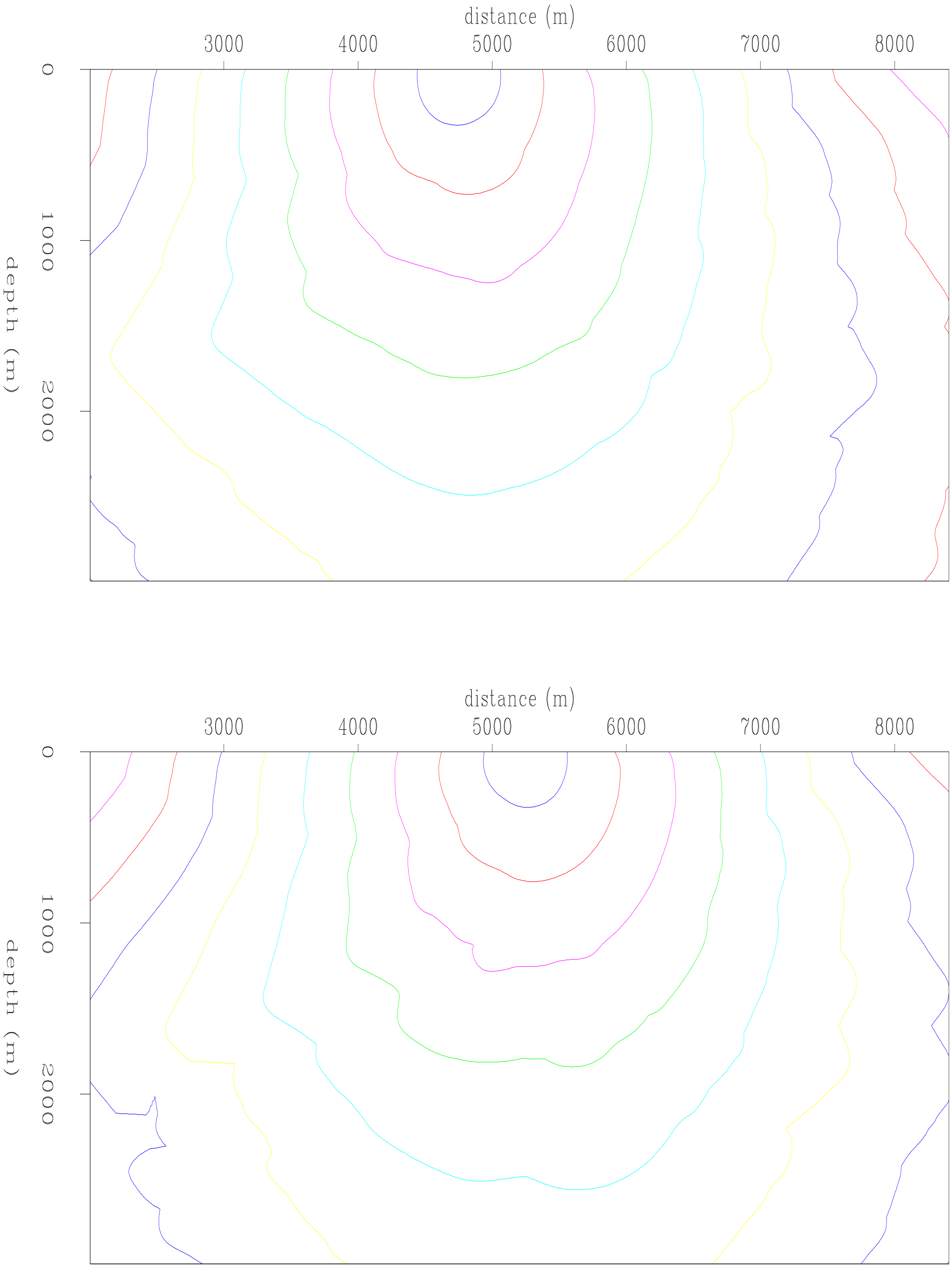


Figure 13: Marmousi velocity model. Contours of traveltime field for the two given surface source positions at 4.75 km (top) and 5.25 km (bottom). david3-m13t [ER]

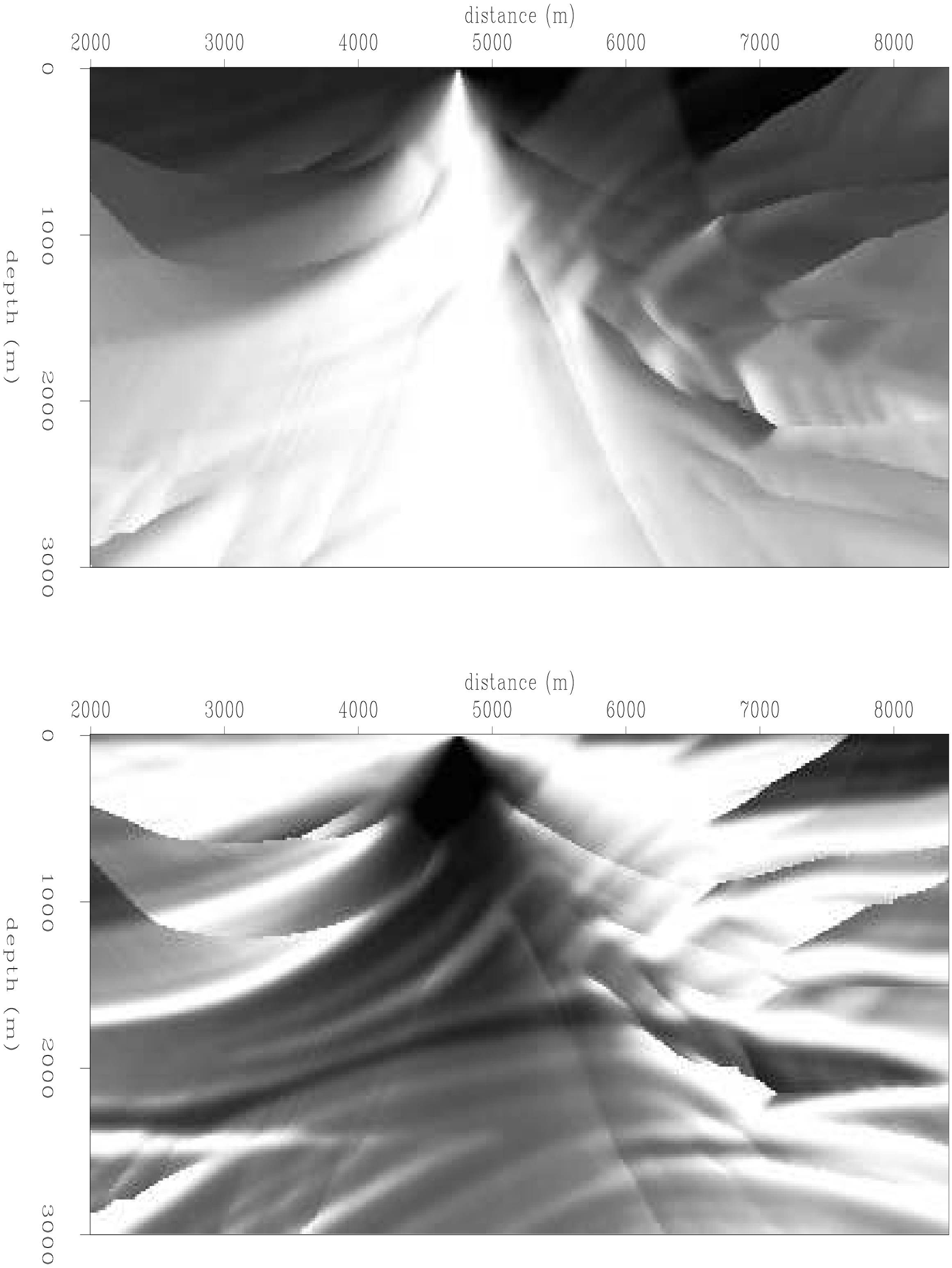


Figure 14: Marmousi velocity model. Horizontal (top) and vertical (bottom) traveltime gradient fields for the surface source positioned at 4.75 km. david3-m1xz [ER]

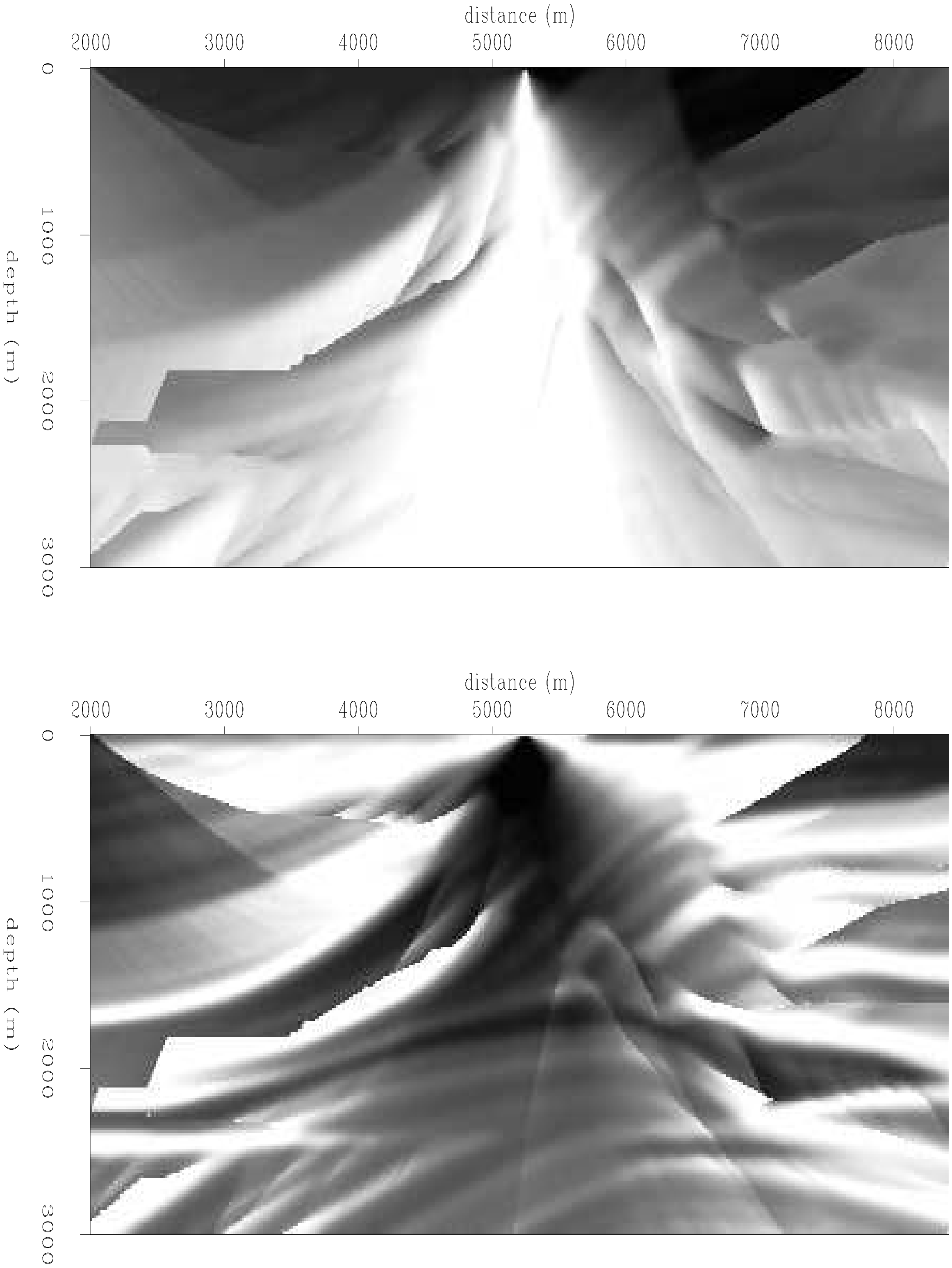
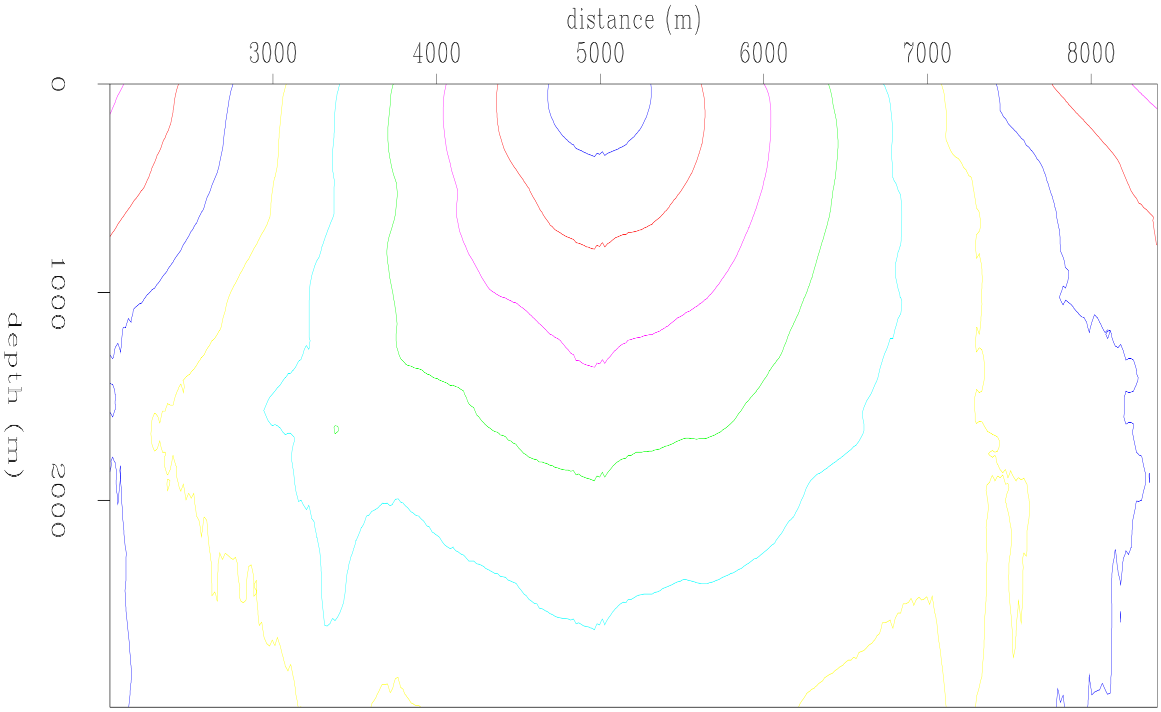


Figure 15: Marmousi velocity model. Horizontal (top) and vertical (bottom) traveltime gradient fields for the surface source positioned at 5.25 km. $\boxed{\text{david3-m3xz}}$ [ER]



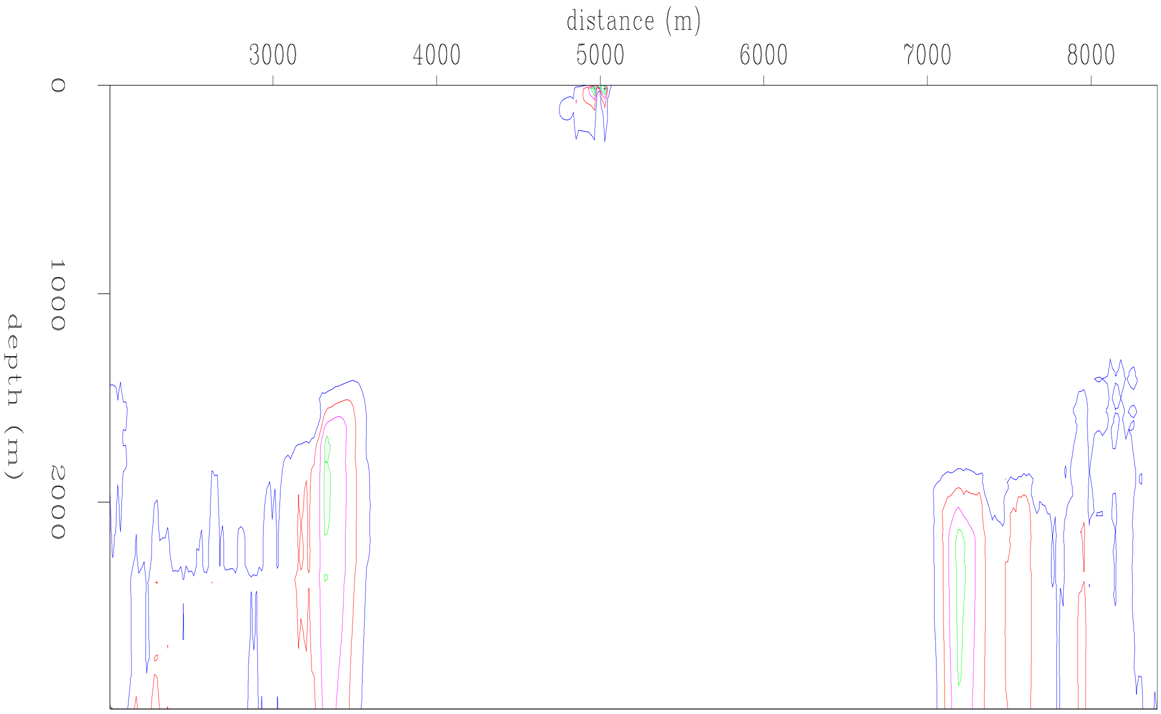
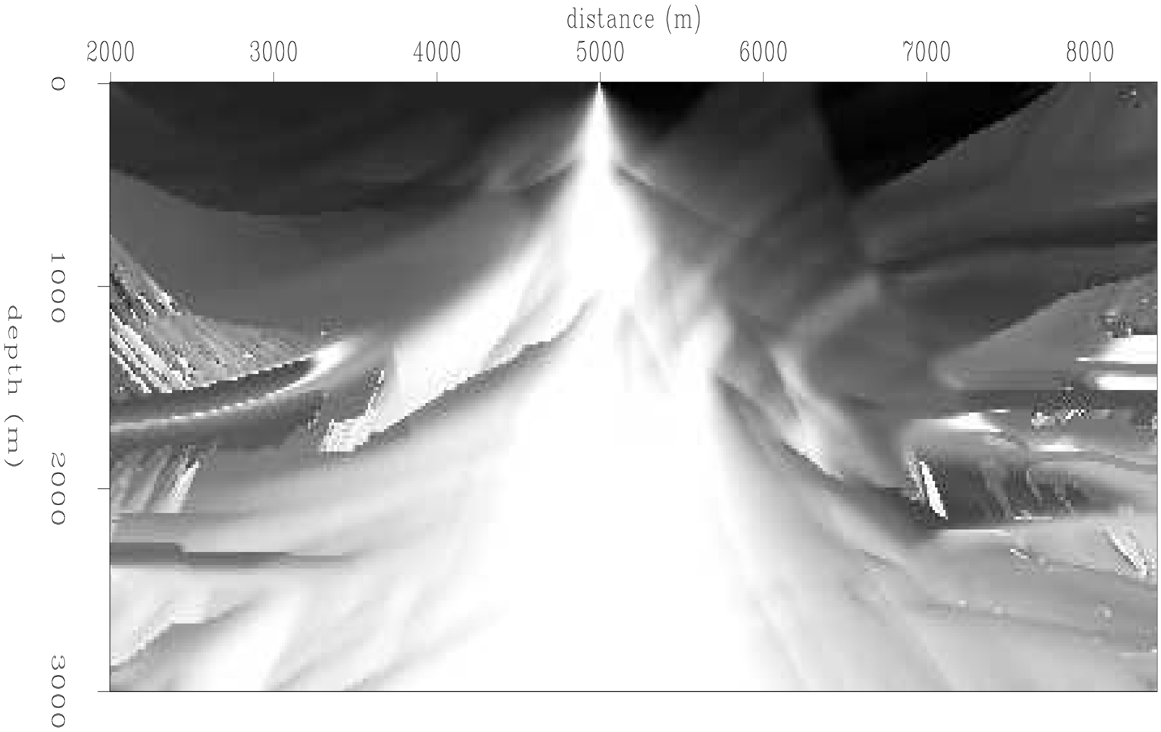


Figure 16: Marmousi velocity model. Interpolated traveltime field. The top panel is the interpolated traveltime field for a surface source positioned at 5.00 km. The lower panel is the relative error between the interpolated traveltime field, and the correct traveltimes modeled by finite differencing the eikonal. The contour farthest from the source region is at 5% relative error, and the contour values increase to 50% error right at the source location, in 5% contour increments. $\boxed{\text{david3-mite}}$ [ER]



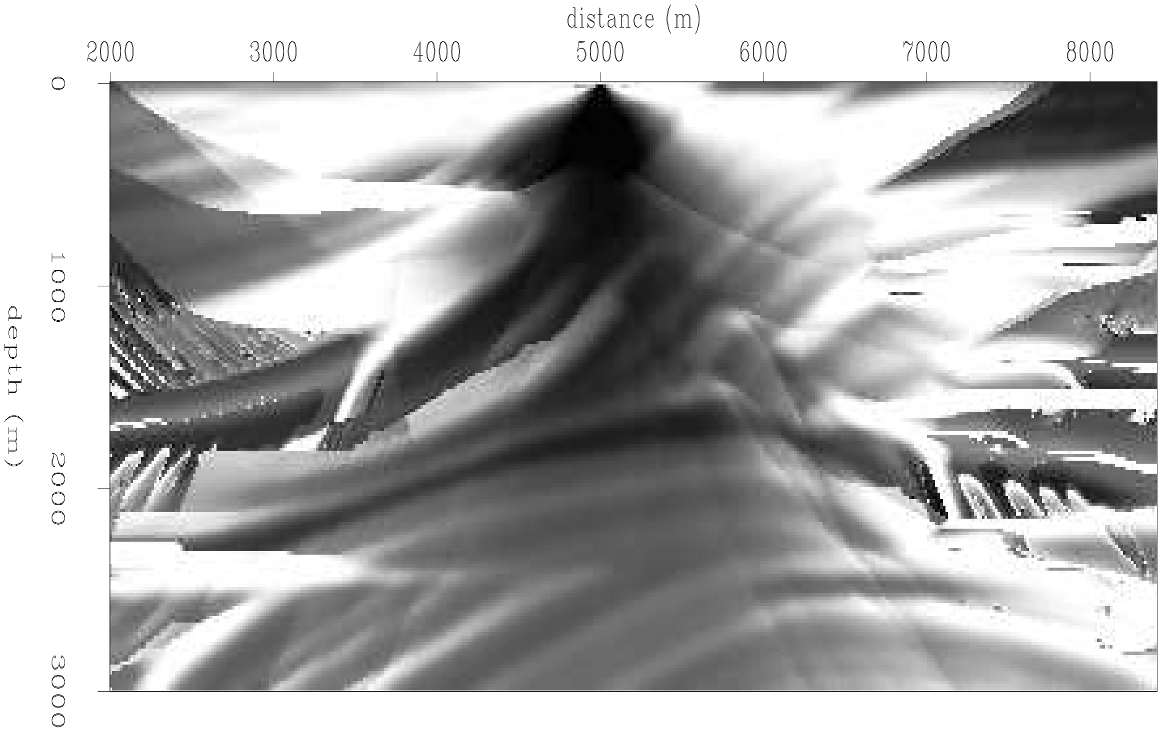


Figure 17: Marmousi velocity model. Horizontal (top) and vertical (bottom) traveltime gradient fields for the interpolated surface source positioned at 5.00 km. $\boxed{\text{david3-mixz}}$ [ER]